CAVA All Hands Meeting
Tues, April 30th 2024
4pm-5pm Pacific | 11am-12pm Melbourne
Location: Foege S448 and zoom

AGENDA:

CountESS with Nick Moore!

CAVA

We build high resolution sequence-function maps for clinically actionable genes

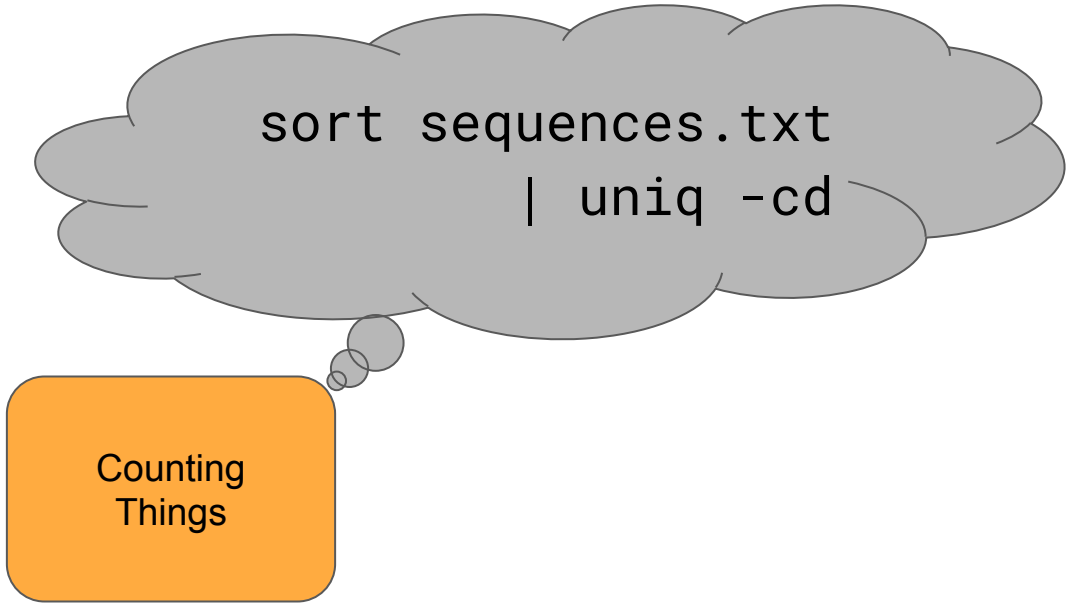
CountESS

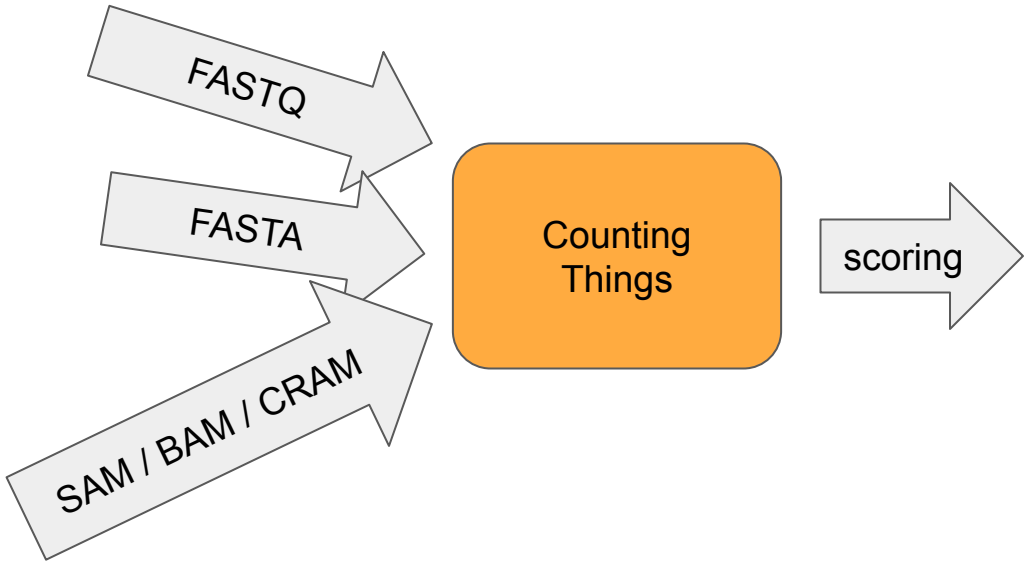
Count-based Experimental Scoring and Statistics

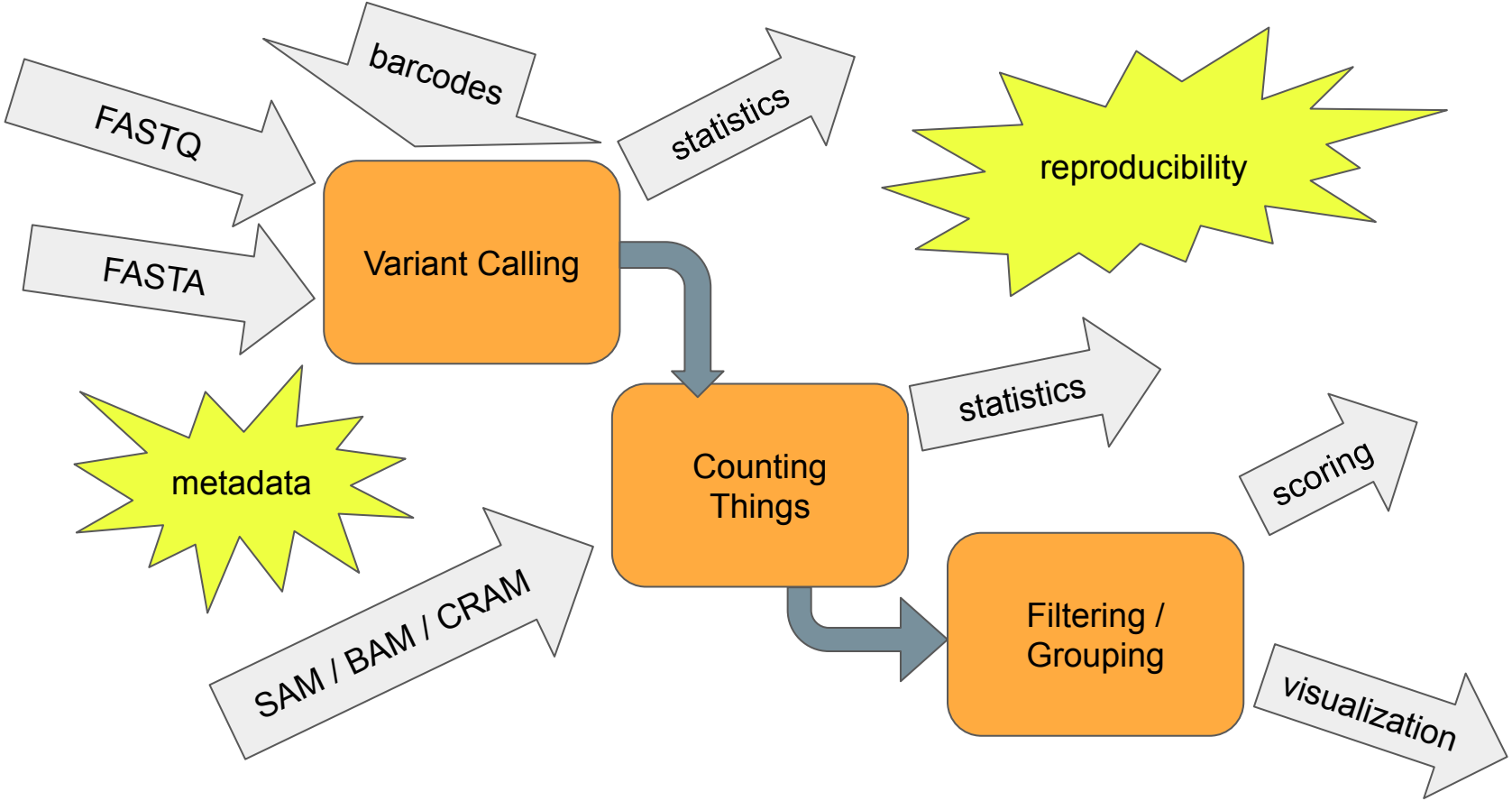
Hollerith 1890 Tabulating Machine and Sorting Box

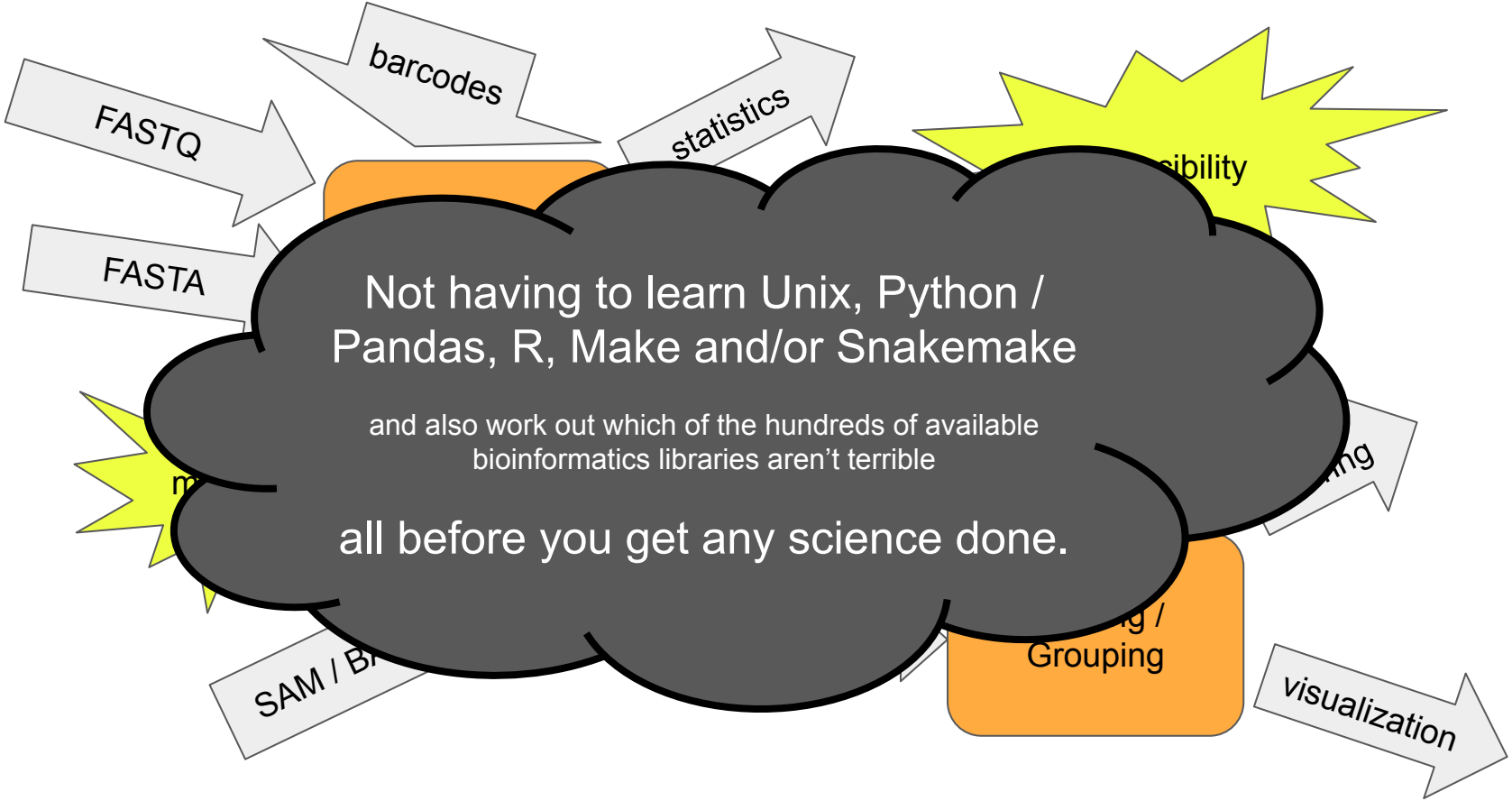
Image: Adam Schuster, CC BY 2.0, via Wikimedia Commons
https://commons.wikimedia.org/wiki/File:HollerithMachine.CHM.jpg


Counting
Things


sort sequences.txt
| uniq -cd
Counting
Things


FASTQ
FASTA
SAM / BAM / CRAM
Counting
Things
scoring


FASTQ
barcodes
statistics
reproducibility
FASTA
Variant Calling
metadata
Counting Things
statistics
scoring
SAM / BAM / CRAM
Filtering / Grouping
visualization


FASTQ
barcodes
statistics
FASTA
Grouping
visualization
Not having to learn Unix, Python / Pandas, R, Make and/or Snakemake
and also work out which of the hundreds of available bioinformatics libraries aren't terrible
all before you get any science done.

Ziemann *et al. Genome Biology* (2016) 17:177
DOI 10.1186/s13059-016-1044-7

Genome Biology

COMMENT Open Access

Gene name errors are widespread in the scientific literature

Mark Ziemann[1], Yotam Eren[1,2] and Assam El-Osta[1,3*]

Abstract

The spreadsheet software Microsoft Excel, when used with default settings, is known to convert gene names to dates and floating-point numbers. A programmatic scan of leading genomics journals reveals that approximately one-fifth of papers with supplementary Excel gene lists contain erroneous gene name conversions.

Keywords: Microsoft Excel, Gene symbol, Supplementary data

Abbreviations: GEO, Gene Expression Omnibus; JIF, journal impact factor

The problem of Excel software (Microsoft Corp., Redmond, WA, USA) inadvertently converting gene symbols to dates and floating-point numbers was originally described in 2004 [1]. For example, gene symbols such as *SEPT2* (Septin 2) and *MARCH1* [Membrane-Associated Ring Finger (C3HC4) 1, E3 Ubiquitin Protein Ligase] are converted by default to '2-Sep' and '1-Mar', respectively. Furthermore, RIKEN identifiers were described to be automatically converted to floating point numbers (i.e. from accession '2310009E13' to '2.31E+13'). Since that report, we have uncovered further instances where gene symbols were converted to dates in supplementary data of recently published papers (e.g. '*SEPT2*' converted to '2006/09/02').

Source: Biomedcentral (emphasis mine)

Enrich2:

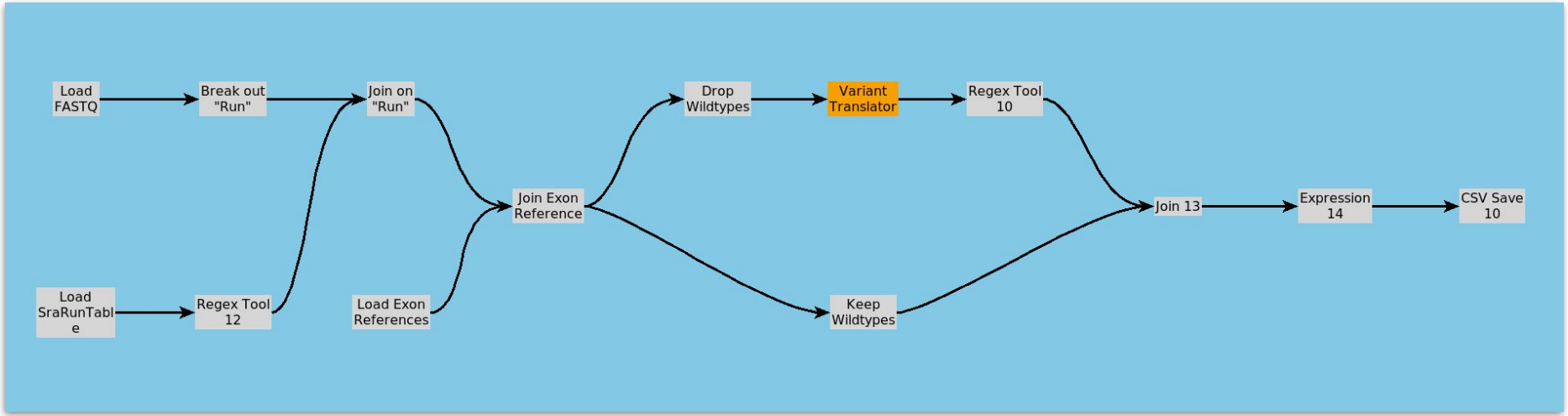
- GUI based
- Python 2 - EOL!
- Specific to a few experiments
- Difficult to extend

CountESS:

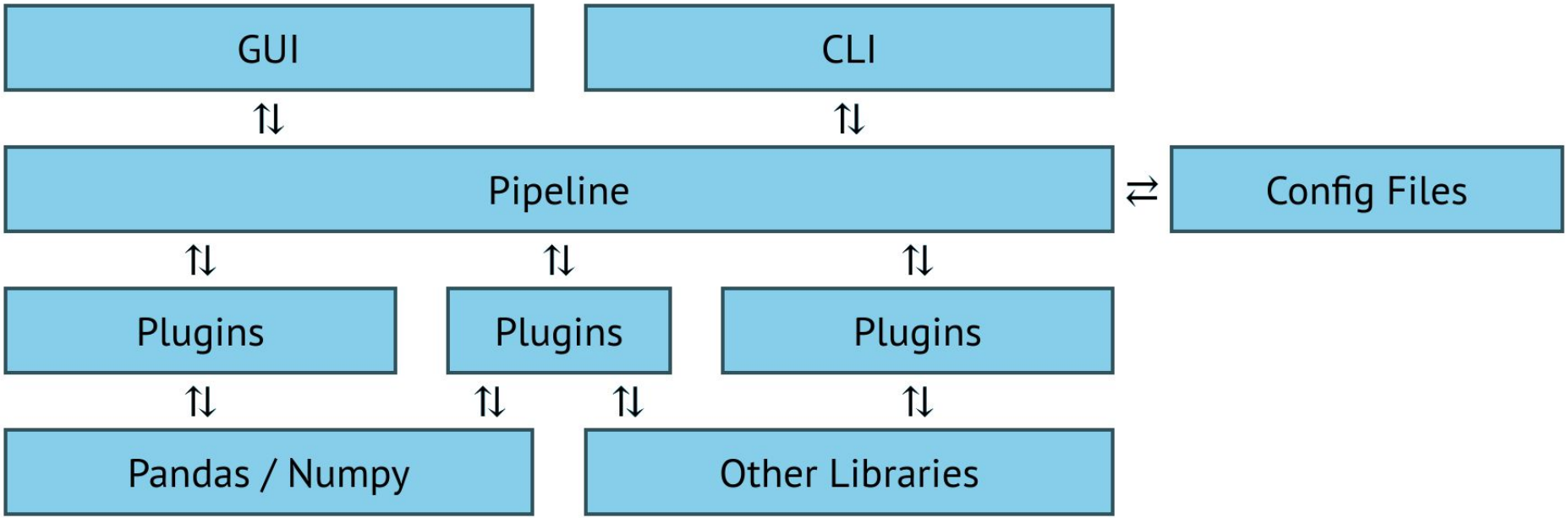
- GUI based
- Python 3.9+
- Plugin architecture
- Simple, single text config file
- Flexible pipeline

CountESS Plugins & Pipelines

- Some plugins included, plus it's easy to install / write new ones.
- Plugins are connected together in a potentially complex pipeline.



CountESS Architecture



CountESS Documentation

https://countess-project.github.io/CountESS/

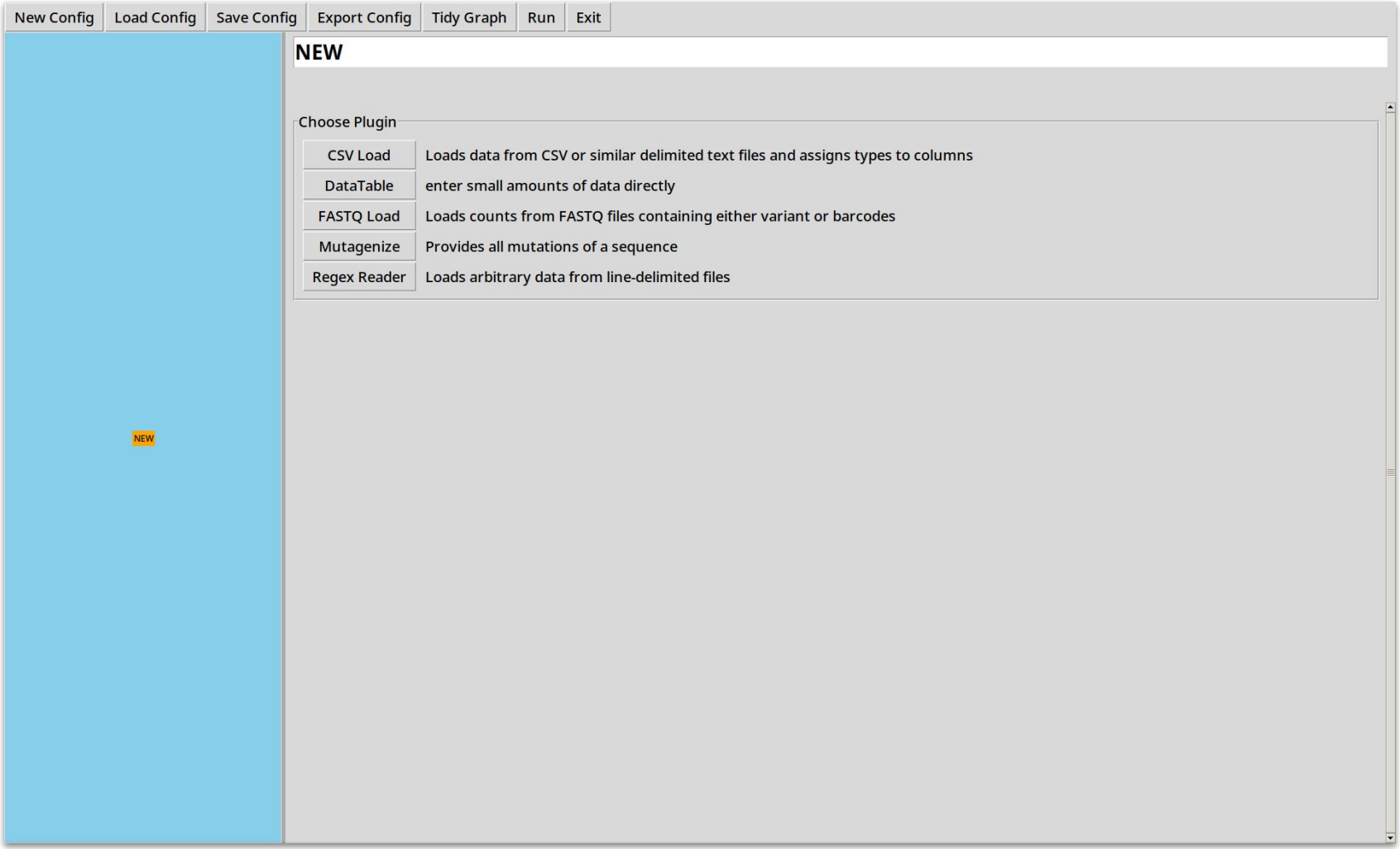
Installing CountESS

```
pip install countess
```

Running CountESS

`countess_gui`

`countess_cmd`


New Config
Load Config
Save Config
Export Config
Tidy Graph
Run
Exit
NEW
NEW
Choose Plugin
CSV Load
Loads data from CSV or similar delimited text files and assigns types to columns
DataTable
enter small amounts of data directly
FASTQ Load
Loads counts from FASTQ files containing either variant or barcodes
Mutagenize
Provides all mutations of a sequence
Regex Reader
Loads arbitrary data from line-delimited files

Examples

https://github.com/CountESS-Project/countess-demo/

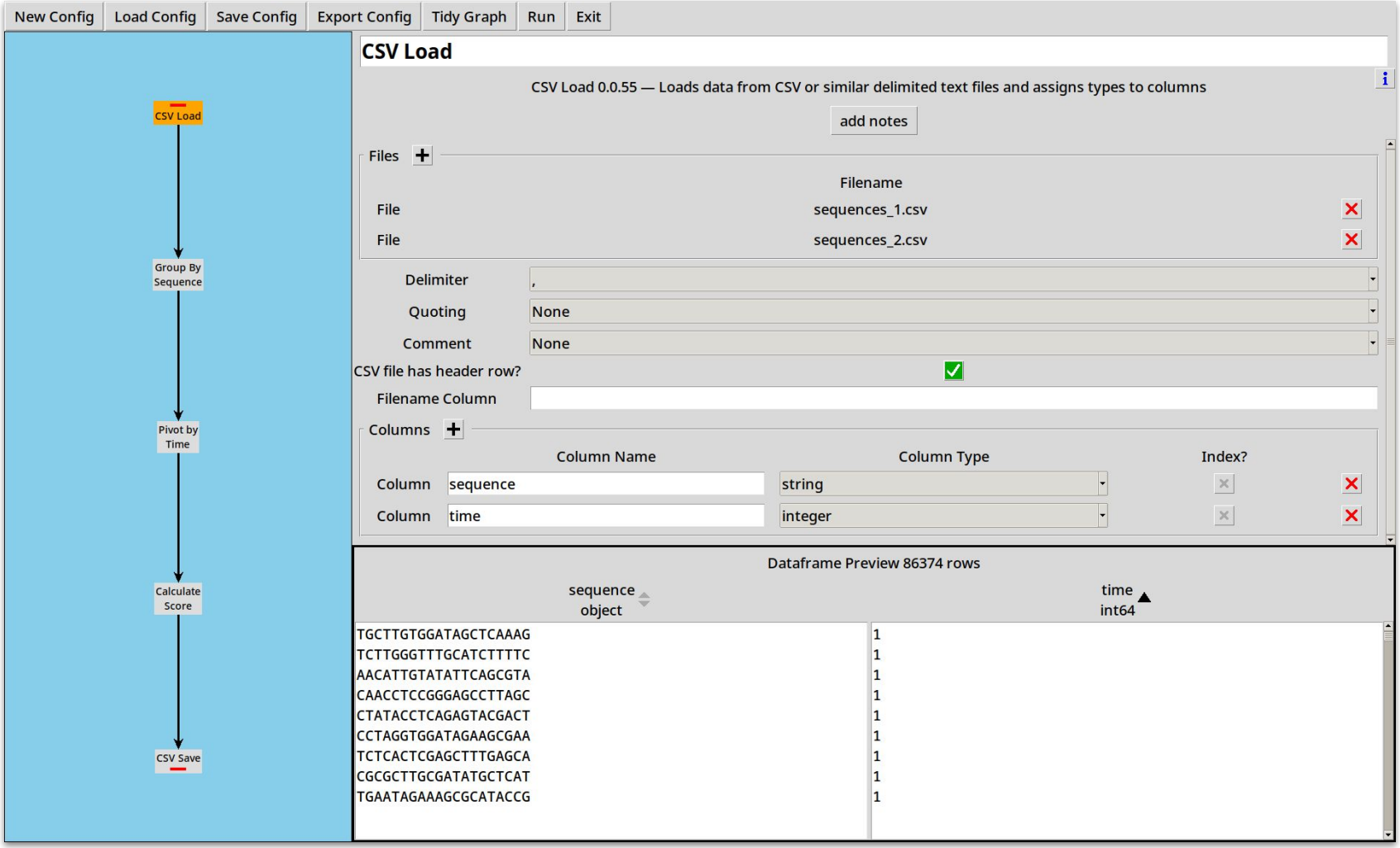
Example 1: Reading & Counting Sequences

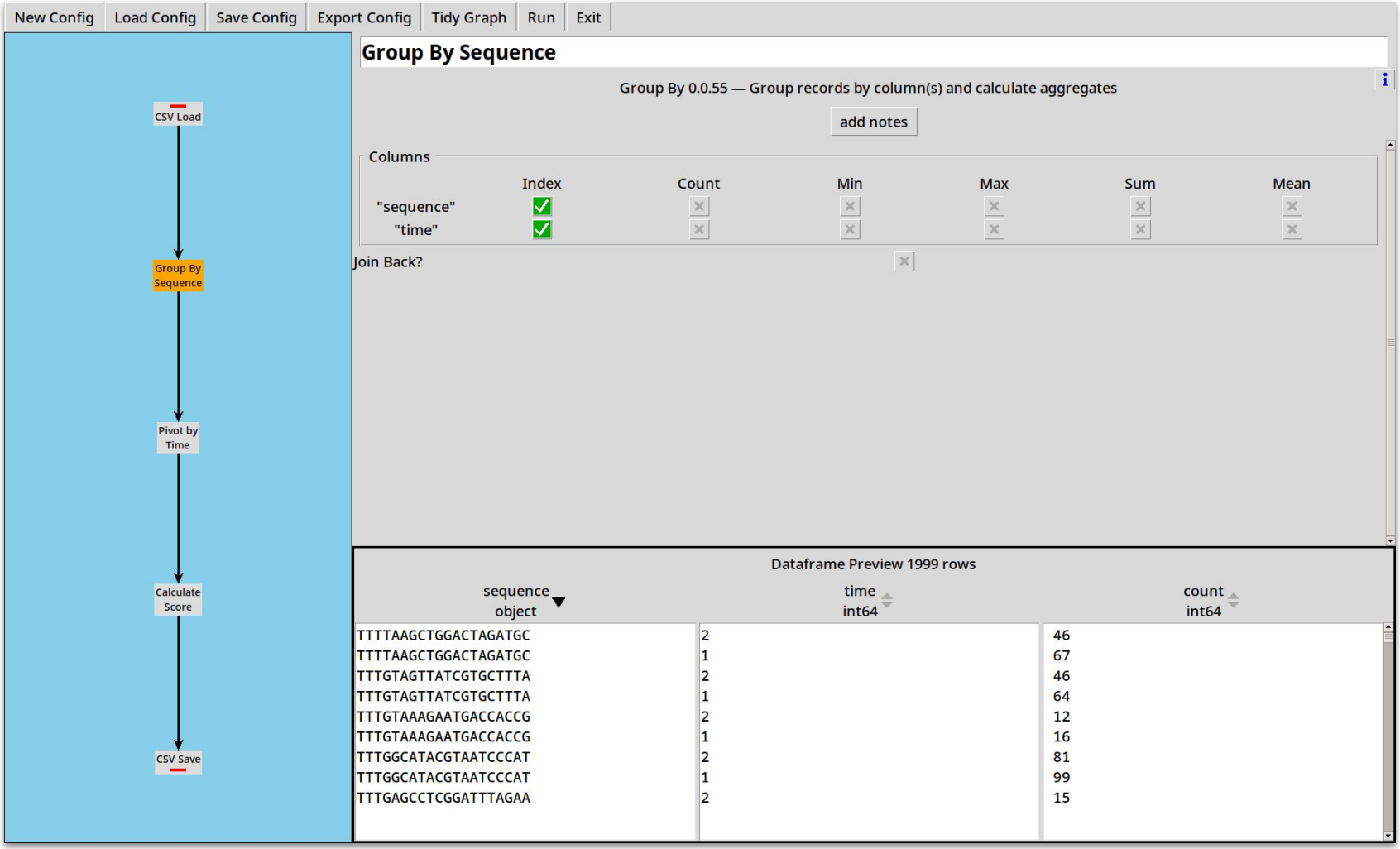

```
sequence,time
AGTTCGAGGACATGGTGAGT,1
GATGTCCTAAGGGTCGATTC,1
TTCAGTACCTAAACTATGTT,1
TTTATATTTCGAGTGAATGT,1
CAACGAGAGATGTAGGAGAA,1
GTAACAGGAGTCATGTTTCC,1
```

(etc)

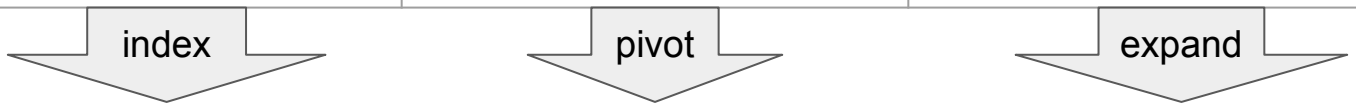
```
sequence,time
GTTCGGCTACCAGGCGAGGA,2
TCTCCTTTGACGACTCGCAC,2
GTTAAGGGCCTCCAAGGAGA,2
AGAGACTGCGCACCGCCCGC,2
GTACACTCAATAAGTACTGT,2
TCATTATACTTCTCAATACT,2
```

(etc)

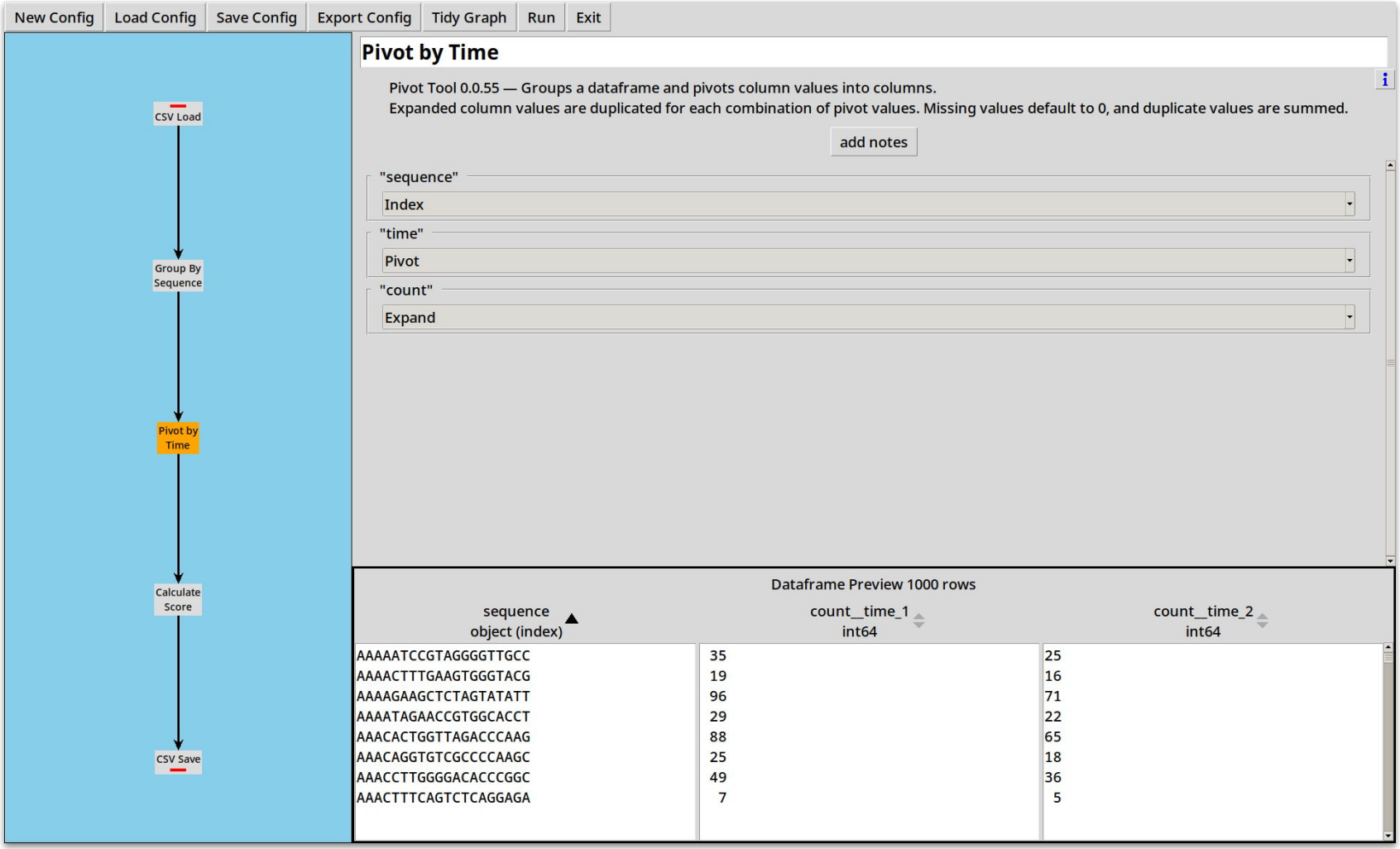

New Config
Load Config
Save Config
Export Config
Tidy Graph
Run
Exit
CSV Load
Group By Sequence
Pivot by Time
Calculate Score
CSV Save
CSV Load
CSV Load 0.0.55 — Loads data from CSV or similar delimited text files and assigns types to columns
add notes
Files
Filename
File
sequences_1.csv
File
sequences_2.csv
Delimiter
,
Quoting
None
Comment
None
CSV file has header row?
Filename Column
Columns
Column Name
Column Type
Index?
Column
sequence
string
Column
time
integer
Dataframe Preview 86374 rows
sequence
object
time
int64
TGCTTGTGGATAGCTCAAAG
TCTTGGGTTTGCATCTTTTC
AACATTGTATATTCAGCGTA
CAACCTCCGGGAGCCTTAGC
CTATACCTCAGAGTACGACT
CCTAGGTGGATAGAAGCGAA
TCTCACTCGAGCTTTGAGCA
CGCGCTTGCGATATGCTCAT
TGAATAGAAAGCGCATACCG
1
1
1
1
1
1
1
1
1
1


New Config
Load Config
Save Config
Export Config
Tidy Graph
Run
Exit
CSV Load
Group By Sequence
Pivot by Time
Calculate Score
CSV Save
Group By Sequence
Group By 0.0.55 — Group records by column(s) and calculate aggregates
add notes
Columns
Index
Count
Min
Max
Sum
Mean
"sequence"
"time"
Join Back?
Dataframe Preview 1999 rows
sequence object
time int64
count int64
TTTTAAGCTGGACTAGATGC 2 46
TTTTAAGCTGGACTAGATGC 1 67
TTTGTAGTTATCGTGCTTTA 2 46
TTTGTAGTTATCGTGCTTTA 1 64
TTTGTAAAGAATGACCACCG 2 12
TTTGTAAAGAATGACCACCG 1 16
TTTGGCATACGTAATCCCAT 2 81
TTTGGCATACGTAATCCCAT 1 99
TTTGAGCCTCGGATTTAGAA 2 15

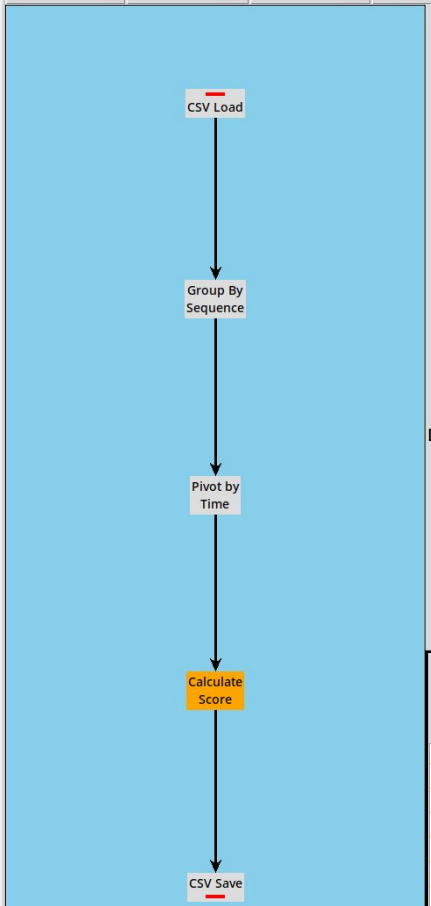
sequence	time	count
AAAAA	1	5
AAAAA	2	7
CCCCC	1	8
GGGGG	1	10
GGGGG	2	3



sequence	count__time_1	count__time_2
AAAAA	5	7
CCCCC	8	0
GGGGG	10	3


New Config
Load Config
Save Config
Export Config
Tidy Graph
Run
Exit
CSV Load
Group By Sequence
Pivot by Time
Calculate Score
CSV Save
Pivot by Time
Pivot Tool 0.0.55 — Groups a dataframe and pivots column values into columns.
Expanded column values are duplicated for each combination of pivot values. Missing values default to 0, and duplicate values are summed.
add notes
"sequence"
Index
"time"
Pivot
"count"
Expand
Dataframe Preview 1000 rows
sequence
object (index)
count__time_1
int64
count__time_2
int64
AAAAATCCGTAGGGGTTGCC 35 25
AAAACTTTGAAGTGGGTACG 19 16
AAAAGAAGCTCTAGTATATT 96 71
AAAATAGAACCGTGGCACCT 29 22
AAACACTGGTTAGACCCAAG 88 65
AAACAGGTGTCGCCCCAAGC 25 18
AAACCTTGGGGACACCCGGC 49 36
AAACTTTCAGTCTCAGGAGA 7 5

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Calculate Score

Python Code 0.0.55 — Apply python code to each row.
Columns are mapped to local variables and back. If you assign to a variable called "__filter", only rows where that value is true will be kept.

add notes

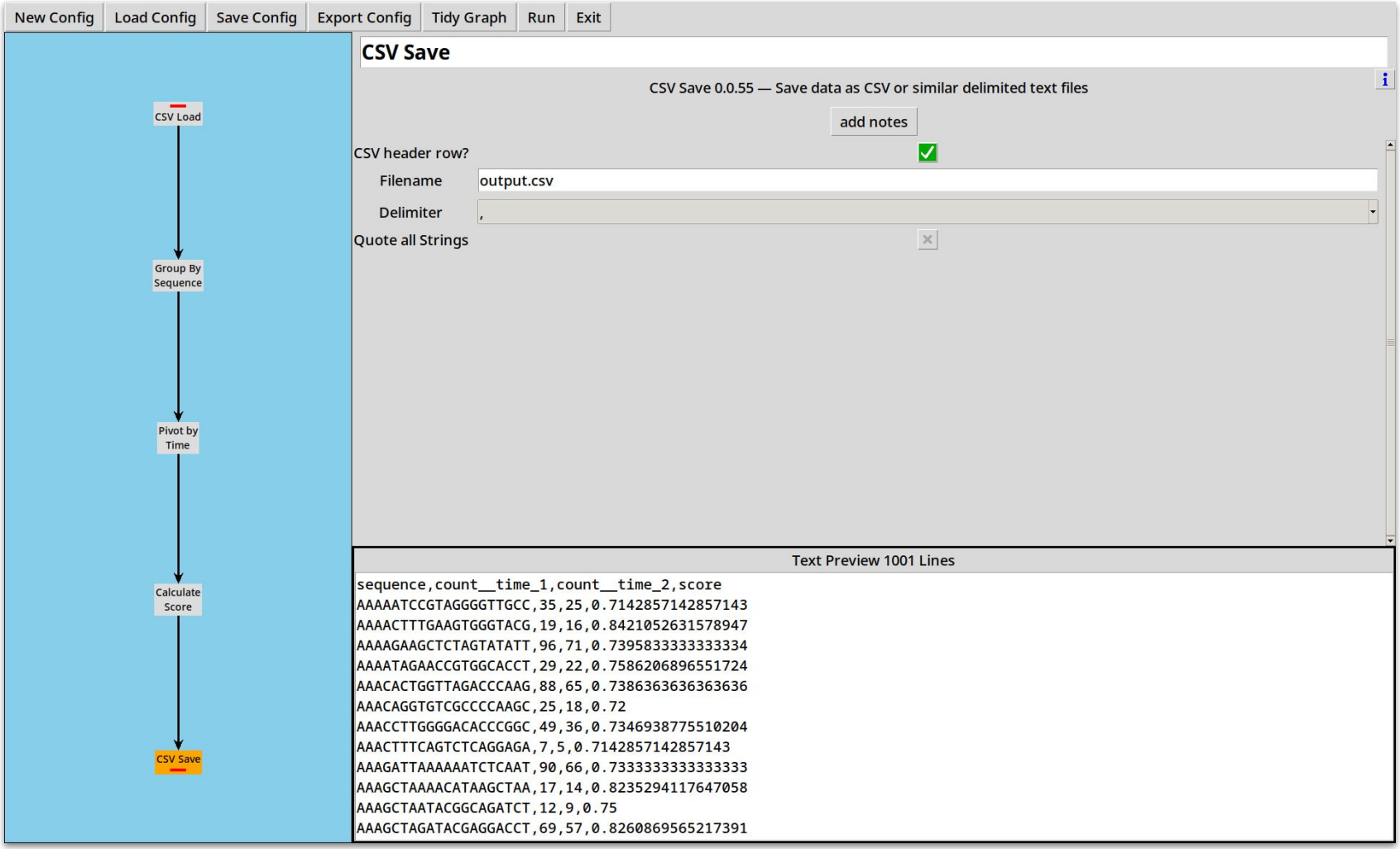
Python Code

```
score = (
    count__time_2 / count__time_1
    if count__time_1 else None
)
```

Drop Null Columns?

Dataframe Preview 1000 rows

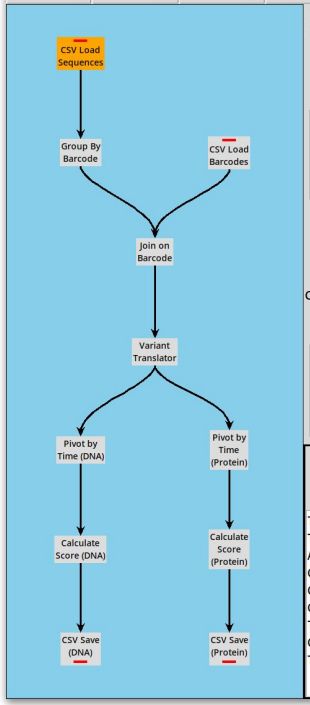
sequence object	count__time_1 int64	count__time_2 int64	score float64
AAAAATCCGTAGGGGTTGCC	35	25	0.714285714286
AAAACTTTGAAGTGGGTACG	19	16	0.842105263158
AAAAGAAGCTCTAGTATATT	96	71	0.739583333333
AAAATAGAACCGTGGCACCT	29	22	0.758620689655
AAACACTGGTTAGACCCAAG	88	65	0.738636363636
AAACAGGTGTCGCCCCAAGC	25	18	0.72
AAACCTTGGGGACACCCGGC	49	36	0.734693877551
AAACTTTCAGTCTCAGGAGA	7	5	0.714285714286


New Config
Load Config
Save Config
Export Config
Tidy Graph
Run
Exit
CSV Load
Group By Sequence
Pivot by Time
Calculate Score
CSV Save
CSV Save
CSV Save 0.0.55 — Save data as CSV or similar delimited text files
add notes
CSV header row?
Filename
output.csv
Delimiter
,
Quote all Strings
Text Preview 1001 Lines
sequence,count__time_1,count__time_2,score
AAAAATCCGTAGGGGTTGCC,35,25,0.7142857142857143
AAAACTTTGAAGTGGGTACG,19,16,0.8421052631578947
AAAAGAAGCTCTAGTATATT,96,71,0.7395833333333334
AAAATAGAACCGTGGCACCT,29,22,0.7586206896551724
AAACACTGGTTAGACCCAAG,88,65,0.7386363636363636
AAACAGGTGTCGCCCCAAGC,25,18,0.72
AAACCTTGGGGACACCCGGC,49,36,0.7346938775510204
AAACTTTCAGTCTCAGGAGA,7,5,0.7142857142857143
AAAGATTAAAAAATCTCAAT,90,66,0.7333333333333333
AAAGCTAAAACATAAGCTAA,17,14,0.8235294117647058
AAAGCTAATACGGCAGATCT,12,9,0.75
AAAGCTAGATACGAGGACCT,69,57,0.8260869565217391

Example 2: Translating Barcodes & Calling Variants

```
barcode,sequence
ATTCCCGTAATCTACGATTA,ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGAT
CCGTCTCCGAGTCACGGTCGAATTTAGGTACTGCACTATCCTTTGAGGCGGGAAGGGCCAC
AAGGGCCGACCCTTGTCGGATAAAATTTGCTAAGAGGAAGGTCTAG
TTACGGTCTGCGTTGGAATC,ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGAT
CCGTCTCCGAGTCACGGTCGAATTTAGGTACTGCACTATCCTTTGAGGCGGGAAGGGCCAC
AAGGGCCGACCCTTGTCGGATAAAATTTGCTAAGAGGAAGGTCTAG
AGGGCCGTGCCAAGTGCAGT,ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGAT
CCGTCTCCGAGTCACGGTCGAATTTAGGTACTGCACTATCCTTTGAGGCGGGAAGGACCAC
AAGGGCCGACCCTTGTCGGATAAAATTTGCTAAGAGGAAGGTCTAG
TGTAGTGCCGTATTTGTGGC,ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGAT
CCGTCTCCGAGTCACGGTCGAATTTAGGTACTGCACTATCCTTTGAGGCAGGAAGGGCCAC
AAGGGCCGACCCTTGTCGGATAAAATTTGCTAAGAGGAAGGTCTAG
```

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



CSV Load Sequences

CSV Load 0.0.55 — Loads data from CSV or similar delimited text files and assigns types to columns

add notes

Files +

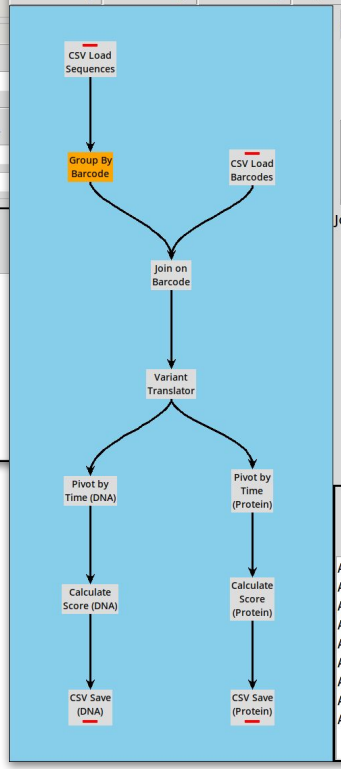
	Filename
File	sequences_1.csv
File	sequences_2.csv

Delimiter ,
Quoting None
Comment None
CSV file has header row?
Filename Column
Columns +

	Column Name
Column	barcode
Column	time

barcode object
TGCTTGTGGATAGCTCAAAG
TCTTGGGTTTGCATCTTTTC
AACATTGTATATTCAGCGTA
CAACCTCCGGGAGCCTTAGC
CTATACCTCAGAGTACGACT
CCTAGGTGGATAGAAGCGAA
TCTCACTCGAGCTTTGAGCA
CGCGCTTGCGATATGCTCAT
TGAATAGAAAGCGCATACCG

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Group By Barcode

Group By 0.0.55 — Group records by column(s) and calculate aggregates

add notes

Columns

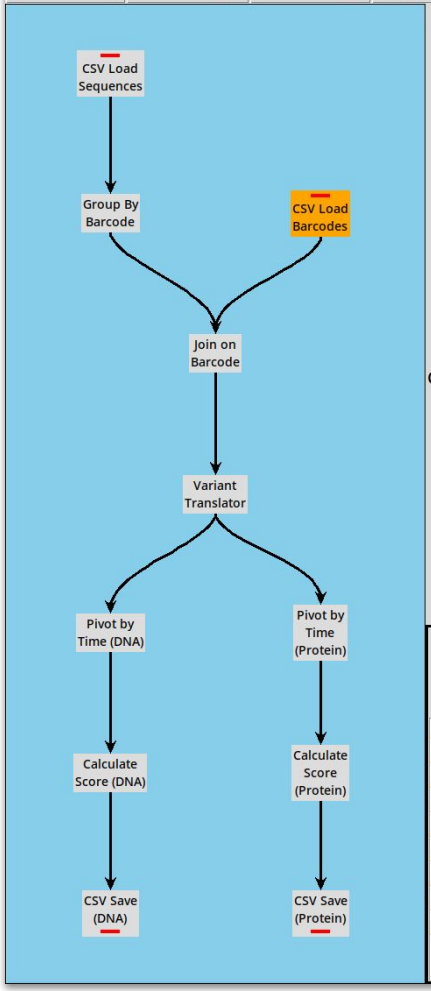
	Index	Count	Min	Max	Sum	Mean
"barcode"	☑	☐	☐	☐	☐	☐
"time"	☑	☐	☐	☐	☐	☐

Join Back? ☐

Dataframe Preview 1999 rows

barcode object (index)	time int64 (index)	count int64
AAAAATCCGTAGGGGTTGCC	1	35
AAAAATCCGTAGGGGTTGCC	2	25
AAAACTTTGAAGTGGGTACG	1	19
AAAACTTTGAAGTGGGTACG	2	16
AAAAGAAGCTCTAGTATATT	1	96
AAAAGAAGCTCTAGTATATT	2	71
AAAATAGAACCGTGGCACCT	1	29
AAAATAGAACCGTGGCACCT	2	22
AAACACTGGTTAGACCCAAG	1	88

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



CSV Load Barcodes

CSV Load 0.0.55 — Loads data from CSV or similar delimited text files and assigns types to columns

add notes

Files +

	Filename
File	barcodes.csv

Delimiter: ,

Quoting: None

Comment: None

CSV file has header row? ☑

Filename Column:

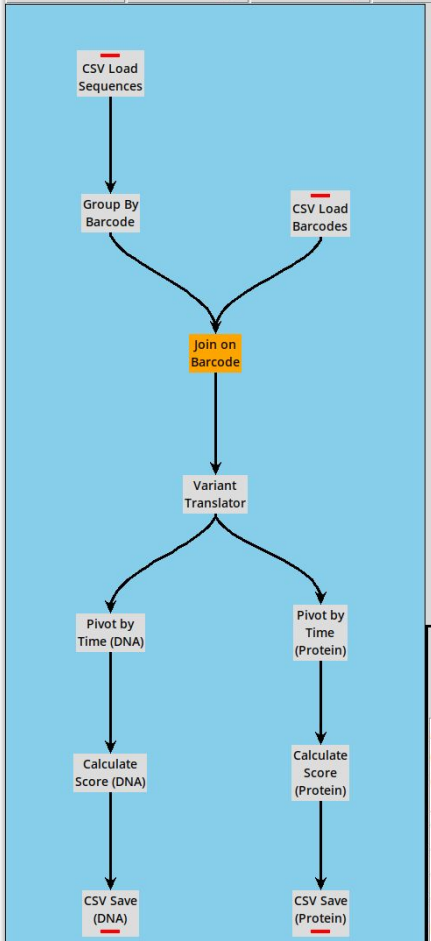
Columns +

	Column Name	Column Type	Index?
Column	barcode	string	
Column	sequence	string	

Dataframe Preview 1000 rows

barcode object	sequence object
AAAAATCCGTAGGGGTTGCC	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC
AAAACTTTGAAGTGGGTACG	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC
AAAAGAAGCTCTAGTATATT	ATGCTTTGTACGGGTGGTGCCCCGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC
AAAATAGAACCGTGGCACCT	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC
AAACACTGGTTAGACCCAAG	ATGCTTTGTACGGGTGGTGCCGTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC
AAACAGGTGTCGCCCCAAGC	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC
AAACCTTGGGGACACCCGGC	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC
AAACTTTCAGTCTCAGGAGA	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC
AAAGATTAAAAAATCTCAAT	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTC

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Join on Barcode

Join 0.0.55 — Joins two Pandas Dataframes by indexes or columns

add notes

Input

Join On: — INDEX —

Required: ☑

Drop Column: ☐

Input

Join On: — INDEX —

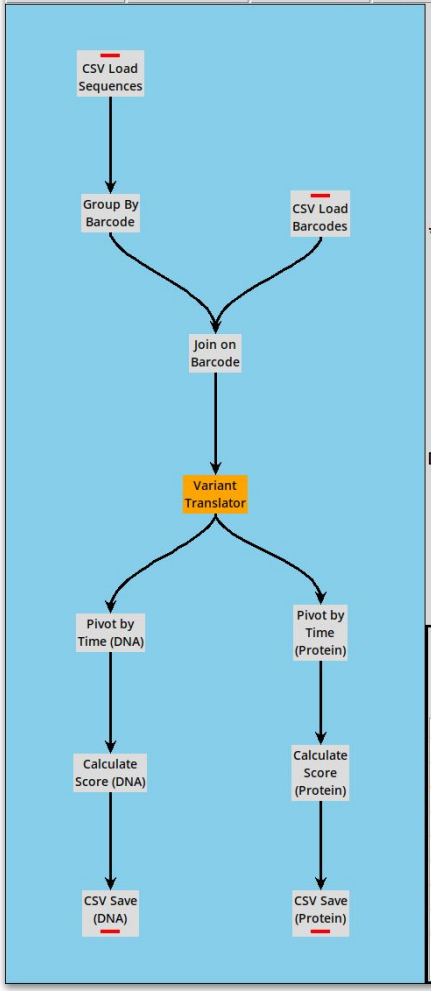
Required: ☑

Drop Column: ☐

Dataframe Preview 1999 rows

barcode object	time int64	count int64	sequence object
ACAGGTATGAGTCTTGTACC	1	53	TTGCTTTGTACGGGTGGTGCCCTGGCTTA
GTGAGTGTATTTGAGATACT	1	96	TTGCTTTGTACGGGTGGTGCCCTGGCTTA
GTGAGTGTATTTGAGATACT	2	74	TTGCTTTGTACGGGTGGTGCCCTGGCTTA
ACAGGTATGAGTCTTGTACC	2	39	TTGCTTTGTACGGGTGGTGCCCTGGCTTA
TAGGTTCCAAATGAAAGATG	1	58	GTGCTTTGTACGGGTGGTGCCCTGGCTTA
TCATTATACTTCTCAATACT	1	49	GTGCTTTGTACGGGTGGTGCCCTGGCTTA
TAGGTTCCAAATGAAAGATG	2	38	GTGCTTTGTACGGGTGGTGCCCTGGCTTA
TCATTATACTTCTCAATACT	2	40	GTGCTTTGTACGGGTGGTGCCCTGGCTTA
CGATGAGAAGGTAGTAAGAT	2	31	CTGCTTTGTACGGGTGGTGCCCTGGCTTA

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Variant Translator

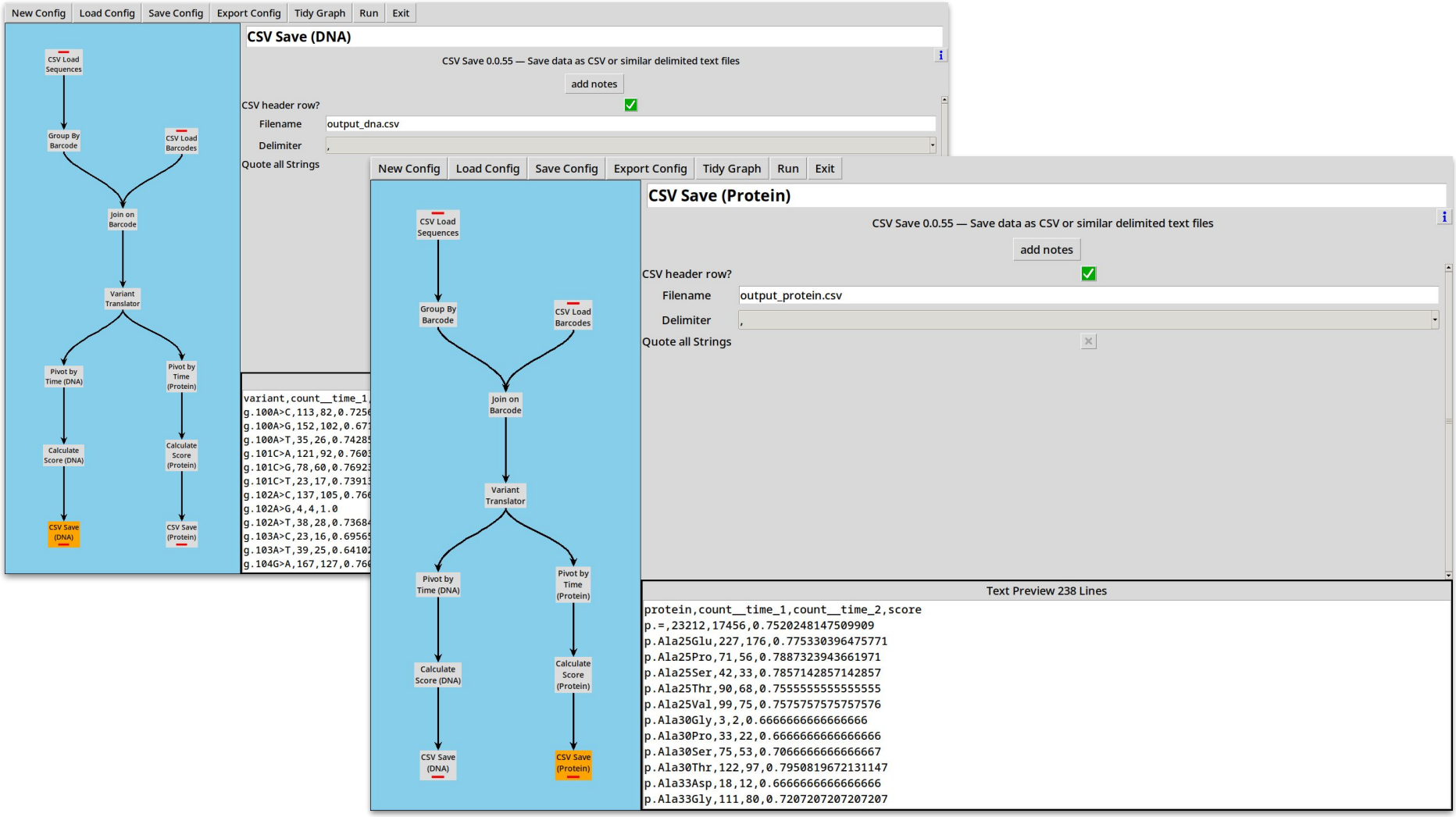
Variant Translator 0.0.55 — Turns a DNA sequence into a HGVS variant code

add notes

Input Column	sequence
Reference Column	— NONE —
OR Reference Sequence	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTCGAATTTAGGTACTGCACTATCCTTTGAGGC
Output Column	variant
Max Mutations	10
Protein Column	protein
Protein Offset	0
Max Protein Variations	10
Drop unidentified variants	
Drop Input Column(s)	

Dataframe Preview 1999 rows

barcode object	time int64	count int64	sequence object	variant object	protein object
AAAAATCCGTAGGGGTTGC	1	35	ATGCTTTGTACGGGTGGTG	g.=	p.=
AAAAATCCGTAGGGGTTGC	2	25	ATGCTTTGTACGGGTGGTG	g.=	p.=
AAAACTTTGAAGTGGGTAC	1	19	ATGCTTTGTACGGGTGGTG	g.138G>A	p.=
AAAACTTTGAAGTGGGTAC	2	16	ATGCTTTGTACGGGTGGTG	g.138G>A	p.=
AAAAGAAGCTCTAGTATAT	1	96	ATGCTTTGTACGGGTGGTG	g.23T>C	p.Leu8Pro
AAAAGAAGCTCTAGTATAT	2	71	ATGCTTTGTACGGGTGGTG	g.23T>C	p.Leu8Pro
AAAATAGAACCGTGGCACC	1	29	ATGCTTTGTACGGGTGGTG	g.113C>T	p.Pro38Leu
AAAATAGAACCGTGGCACC	2	22	ATGCTTTGTACGGGTGGTG	g.113C>T	p.Pro38Leu
AAACACTGGTTAGACCCAA	1	88	ATGCTTTGTACGGGTGGTG	g.22C>G	p.Leu8Val


New Config
Load Config
Save Config
Export Config
Tidy Graph
Run
Exit
CSV Load Sequences
Group By Barcode
CSV Load Barcodes
Join on Barcode
Variant Translator
Pivot by Time (DNA)
Pivot by Time (Protein)
Calculate Score (DNA)
Calculate Score (Protein)
CSV Save (DNA)
CSV Save (Protein)
CSV Save (DNA)
CSV Save 0.0.55 — Save data as CSV or similar delimited text files
add notes
CSV header row?
Filename
output_dna.csv
Delimiter
,
Quote all Strings
variant,count__time_1,
g.100A>C,113,82,0.725
g.100A>G,152,102,0.67
g.100A>T,35,26,0.74285
g.101C>A,121,92,0.7603
g.101C>G,78,60,0.76923
g.101C>T,23,17,0.73913
g.102A>C,137,105,0.766
g.102A>G,4,4,1.0
g.102A>T,38,28,0.73684
g.103A>C,23,16,0.69565
g.103A>T,39,25,0.64102
g.104G>A,167,127,0.760
CSV Save (Protein)
CSV Save 0.0.55 — Save data as CSV or similar delimited text files
add notes
CSV header row?
Filename
output_protein.csv
Delimiter
,
Quote all Strings
Text Preview 238 Lines
protein,count__time_1,count__time_2,score
p.=,23212,17456,0.752024814750990
p.Ala25Glu,227,176,0.775330396475771
p.Ala25Pro,71,56,0.7887323943661971
p.Ala25Ser,42,33,0.7857142857142857
p.Ala25Thr,90,68,0.7555555555555555
p.Ala25Val,99,75,0.7575757575757576
p.Ala30Gly,3,2,0.6666666666666666
p.Ala30Pro,33,22,0.6666666666666666
p.Ala30Ser,75,53,0.7066666666666667
p.Ala30Thr,122,97,0.7950819672131147
p.Ala33Asp,18,12,0.6666666666666666
p.Ala33Gly,111,80,0.7207207207207207

Example 3: FASTQ and VAMP-seq

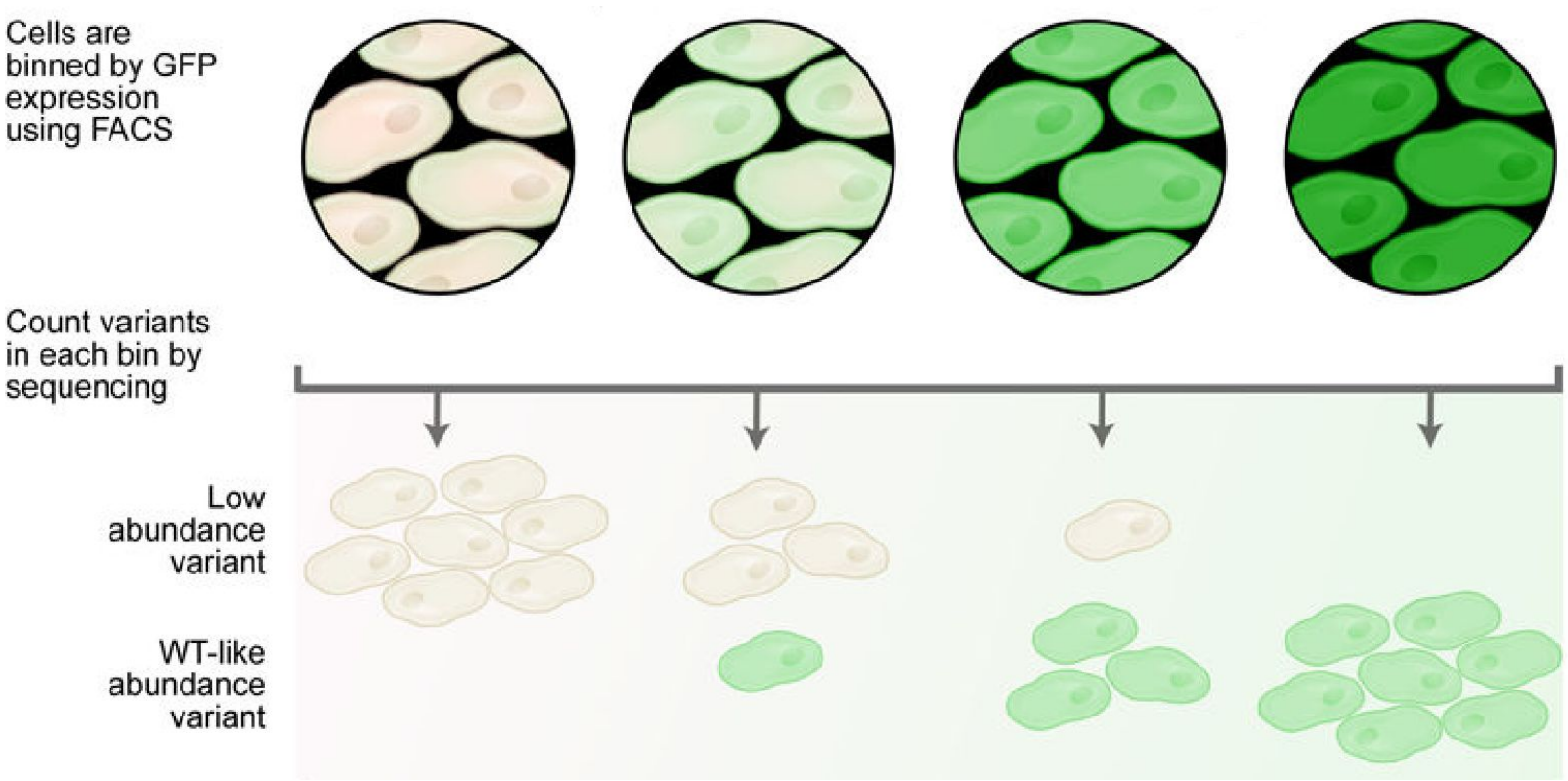
vampseq_1_1.fastq

```
@COUNTESS-TEST-DATA:0
GGTAGAAGATAAAAAAGCA
+COUNTESS-TEST-DATA:0
ZZZZXYZZYXYZZYXYZZYX
@COUNTESS-TEST-DATA:1
GCTACTACGTGGACTGGCCA
+COUNTESS-TEST-DATA:1
ZZZZXYZZYXYZZYXYZZYX
@COUNTESS-TEST-DATA:2
AAGGGATCGAAAGTCCCAAT
+COUNTESS-TEST-DATA:2
ZZZZXYZZYXYZZYXYZZYX
```

vampseq_1_2.fastq

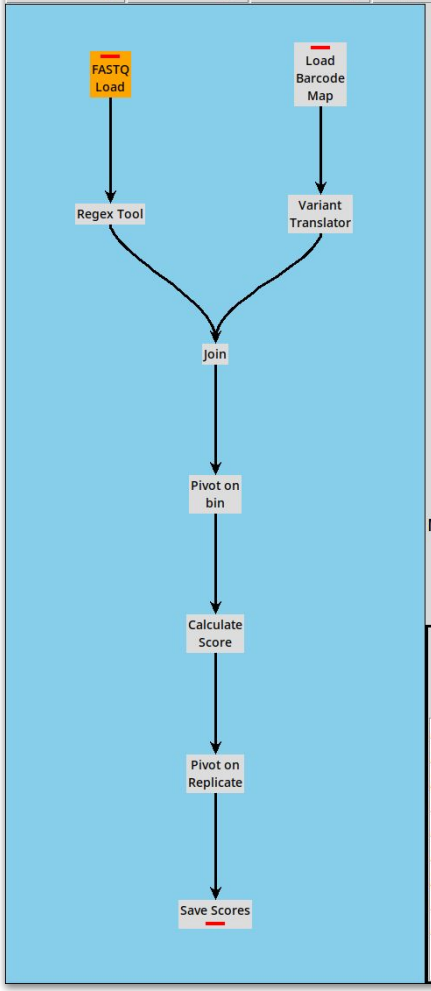
```
@COUNTESS-TEST-DATA:0
CTATATGGGCATTCCCGACT
+COUNTESS-TEST-DATA:0
ZZZZXYZZYXYZZYXYZZYX
@COUNTESS-TEST-DATA:1
TGGCAGATACGGTCAAGTCA
+COUNTESS-TEST-DATA:1
ZZZZXYZZYXYZZYXYZZYX
@COUNTESS-TEST-DATA:2
GAGGACCTTGTCTTTGAAAG
+COUNTESS-TEST-DATA:2
ZZZZXYZZYXYZZYXYZZYX
```

Example 3: FASTQ and VAMP-seq



Reference: "Multiplex Assessment of Protein Variant Abundance by Massively Parallel Sequencing"

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



FASTQ Load

FASTQ Load 0.0.55 — Loads counts from FASTQ files containing either variant or barcodes

add notes

Files +

	Filename	
File	vampseq_1_1.fastq	✕
File	vampseq_1_2.fastq	✕
File	vampseq_2_1.fastq	✕
File	vampseq_2_2.fastq	✕
File	vampseq_3_1.fastq	✕
File	vampseq_3_2.fastq	✕
File	vampseq_4_1.fastq	✕
File	vampseq_4_2.fastq	✕

Minimum Average Quality: 10.0

Header Column? ☐

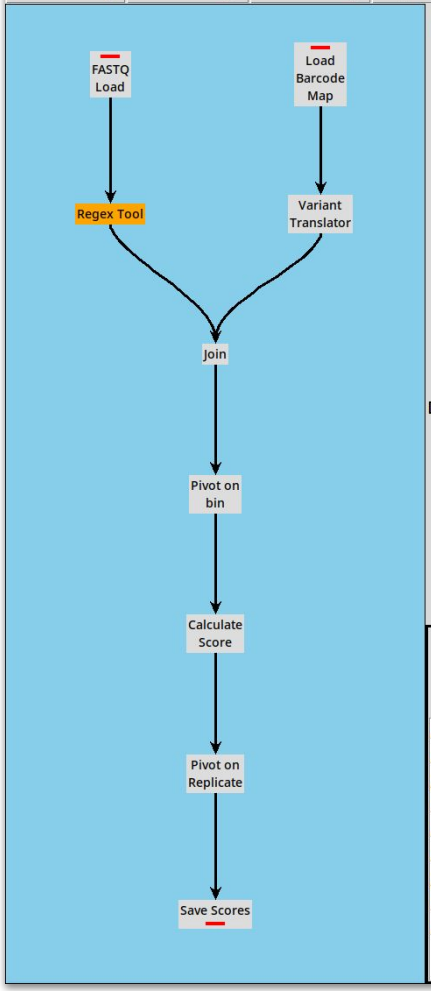
Filename Column? ☑

Group by Sequence? ☑

Dataframe Preview 7563 rows

sequence object (index)	filename object (index)	count int64
AAAAATCCGTAGGGGTTGCC	vampseq_1_1.fastq	8
AAAAATCCGTAGGGGTTGCC	vampseq_1_2.fastq	9
AAAAATCCGTAGGGGTTGCC	vampseq_2_1.fastq	16
AAAAATCCGTAGGGGTTGCC	vampseq_2_2.fastq	17
AAAAATCCGTAGGGGTTGCC	vampseq_3_1.fastq	14
AAAAATCCGTAGGGGTTGCC	vampseq_3_2.fastq	22
AAAAATCCGTAGGGGTTGCC	vampseq_4_1.fastq	9
AAAAATCCGTAGGGGTTGCC	vampseq_4_2.fastq	9
AAAACTTTGAAGTGGGTACG	vampseq_1_1.fastq	4

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Regex Tool

Regex Tool 0.0.55 — Apply regular expressions to a column to make new column(s)

add notes

Input Column: filename

Regular Expression: vampseq_(\d)_(\d)\.fastq

Output Columns +

	Column Name	Column Type
Col	bin	integer
Col	rep	integer

Drop Column

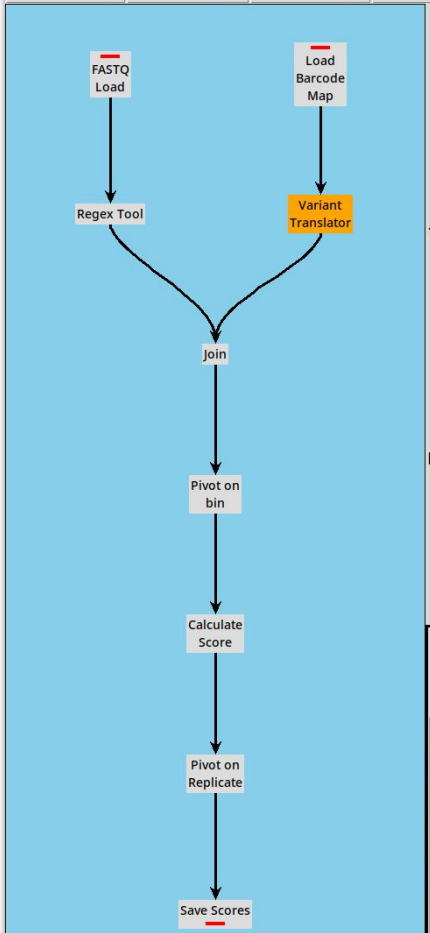
Drop Unmatched Rows

Multi Match

Dataframe Preview 7563 rows

sequence object (index)	filename object (index)	count int64	bin int64	rep int64
AAAAATCCGTAGGGGTTGCC	vampseq_1_1.fastq	8	1	1
AAAAATCCGTAGGGGTTGCC	vampseq_1_2.fastq	9	1	2
AAAAATCCGTAGGGGTTGCC	vampseq_2_1.fastq	16	2	1
AAAAATCCGTAGGGGTTGCC	vampseq_2_2.fastq	17	2	2
AAAAATCCGTAGGGGTTGCC	vampseq_3_1.fastq	14	3	1
AAAAATCCGTAGGGGTTGCC	vampseq_3_2.fastq	22	3	2
AAAAATCCGTAGGGGTTGCC	vampseq_4_1.fastq	9	4	1
AAAAATCCGTAGGGGTTGCC	vampseq_4_2.fastq	9	4	2
AAAACTTTGAAGTGGGTACG	vampseq_1_1.fastq	4	1	1

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Variant Translator

Variant Translator 0.0.55 — Turns a DNA sequence into a HGVS variant code

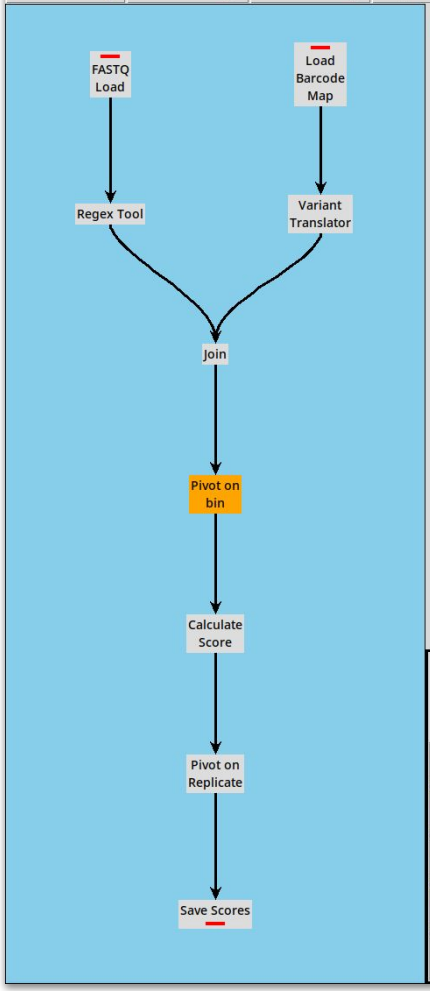
add notes

Input Column	sequence
Reference Column	— NONE —
OR Reference Sequence	ATGCTTTGTACGGGTGGTGCCCTGGCTTATCTATCTAGATCCGTCTCCGAGTCACGGTCGAATTTAGGTACTGCACTATCCTTTGAGGC
Output Column	variant
Max Mutations	10
Protein Column	protein
Protein Offset	0
Max Protein Variations	10
Drop unidentified variants	
Drop Input Column(s)	

Dataframe Preview 1000 rows

barcode object	sequence object	variant object	protein object
CGTGTCTATTTGCACCATCA	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.100A>C	p.Thr34Pro
GGAGCATAGGTGGAGCTACA	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.100A>C	p.Thr34Pro
CGAGTCGTCGTTCTTGCGTA	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.100A>G	p.Thr34Ala
GCCAACAACAGACAAGATGG	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.100A>G	p.Thr34Ala
CATCATTTCAACTGAACCTC	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.100A>G	p.Thr34Ala
CCCTGACCTCAATCCTTCTG	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.100A>T	p.Thr34Ser
AAGACGTAAGAGCTTGTCAC	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.101C>A	p.Thr34Lys
TGTACCCACCCACGCCGTCG	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.101C>A	p.Thr34Lys
CTTAACTTCAGTTCTGCTAG	ATGCTTTGTACGGGTGGTGCCCTGGCTTA	g.101C>A	p.Thr34Lys

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Pivot on bin

Pivot Tool 0.0.55 — Groups a dataframe and pivots column values into columns.
Expanded column values are duplicated for each combination of pivot values. Missing values default to 0, and duplicate values are summed.

add notes

"sequence_x": Drop

"filename": Drop

"count": Expand

"bin": Pivot

"rep": Index

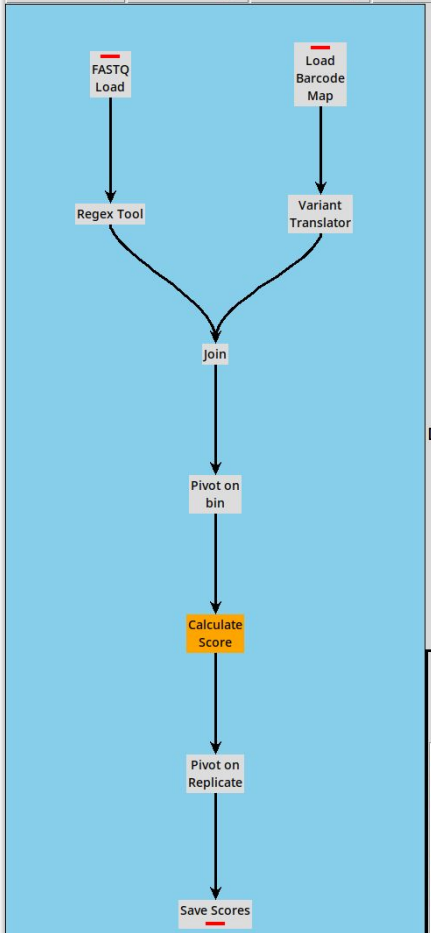
"sequence_y": Drop

"variant": Drop

Dataframe Preview 474 rows

rep int64 (index)	protein object (index)	count_bin_1 float64	count_bin_2 float64	count_bin_3 float64	count_bin_4 float64
1	p.=	6099.	5783.	5641.	5068.
2	p.=	5936.	5898.	5537.	5145.
1	p.Ala25Glu	29.	40.	60.	66.
2	p.Ala25Glu	23.	42.	61.	73.
1	p.Ala25Pro	42.	15.	6.	2.
2	p.Ala25Pro	25.	17.	9.	0.
1	p.Ala25Ser	25.	22.	18.	42.
2	p.Ala25Ser	43.	23.	27.	37.

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Calculate Score

Python Code 0.0.55 — Apply python code to each row.
Columns are mapped to local variables and back. If you assign to a variable called "__filter", only rows where that value is true will be kept.

add notes

Python Code

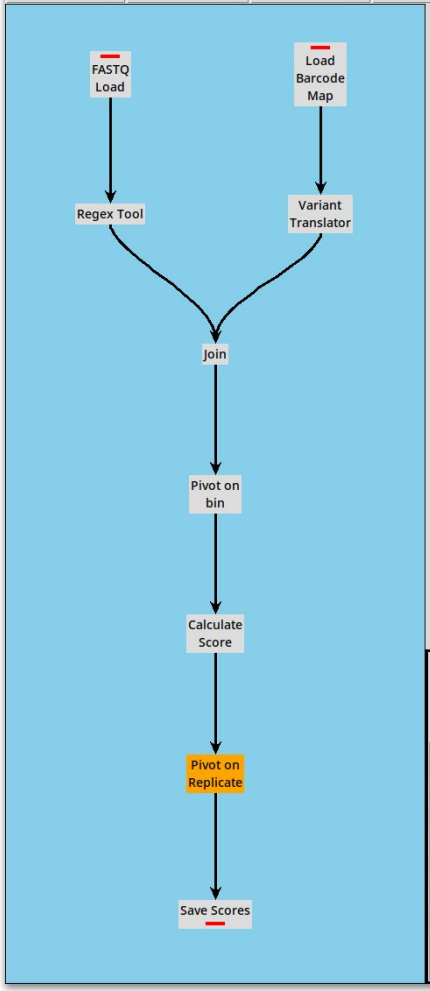
```
score = (0.25 * count__bin_1 + 0.5 * count__bin_2 + 0.75 * count__bin_3 + count__bin_4) / (count__bi
n_1 + count__bin_2 + count__bin_3 + count__bin_4)
```

Drop Null Columns?

Dataframe Preview 474 rows

rep int64	protein object	count__bin_1 float64	count__bin_2 float64	count__bin_3 float64	count__bin_4 float64	score float64
1	p.=	6099.	5783.	5641.	5068.	0.607100172635
1	p.Lys47Met	5.	9.	21.	19.	0.75
1	p.Lys47_Val48del	68.	49.	35.	15.	0.495508982036
1	p.Met1Arg	6.	11.	18.	11.	0.684782608696
1	p.Met1Ile	97.	74.	47.	58.	0.559782608696
1	p.Met1Leu	15.	40.	66.	94.	0.777906976744
1	p.Met1Lys	12.	25.	13.	1.	0.514705882353
1	p.Met1Thr	42.	39.	42.	20.	0.56993006993

New Config | Load Config | Save Config | Export Config | Tidy Graph | Run | Exit



Pivot on Replicate

Pivot Tool 0.0.55 — Groups a dataframe and pivots column values into columns.
Expanded column values are duplicated for each combination of pivot values. Missing values default to 0, and duplicate values are summed.

add notes

"rep" — Pivot

"protein" — Index

"count_bin_1" — Drop

"count_bin_2" — Drop

"count_bin_3" — Drop

"count_bin_4" — Drop

"score" — Expand

Dataframe Preview 237 rows

protein object (index)	score_rep_1 float64	score_rep_2 float64
p.=	0.607100172635	0.609821904424
p.Ala25Glu	0.708974358974	0.731155778894
p.Ala25Pro	0.376923076923	0.421568627451
p.Ala25Ser	0.679906542056	0.611538461538
p.Ala25Thr	0.665816326531	0.618279569892
p.Ala25Val	0.545698924731	0.5225
p.Ala30Gly	0.34375	0.322916666667
p.Ala30Pro	0.34756097561	0.331818181818

CountESS Status

Currently v0.0.61
About to be v0.1.0

Working with several datasets and performing quite well.

Still needs feedback and I’m available to help!

CountESS Future

Reading paired sequences.

Reading SRA datasets directly!

Simple visualizations.

More efficient processing for large data files.