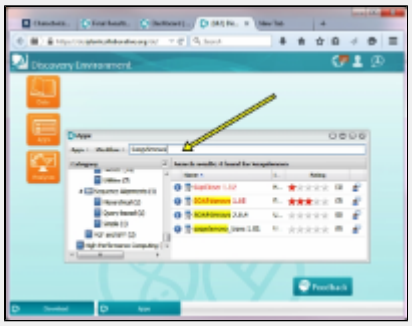
mini SOAPdenovo (Tutorial)

Alert

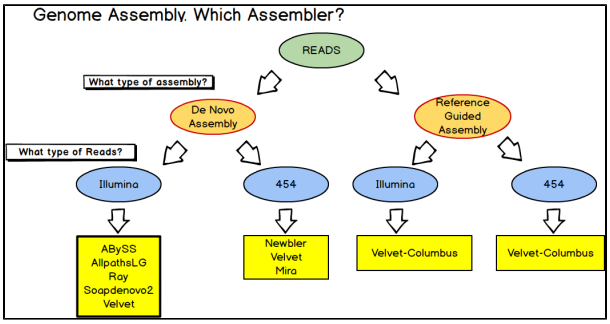


The iPlant App Store is currently being restructured, and apps are being moved to an HPC environment. During this transition, users may occasionally be unable to locate or use apps that are listed in our tutorials. In many cases, these apps can be located by searching them using the search bar at the top of the Apps window in the DE. To increase the chance for search success, try not searching the entire app name and version number but only the portion that refers to the app's function or origin (e.g. 'SOAPdenovo' instead of 'SOAPdenovo-Trans 1.01'). In critical cases, please report your concern to the iPlant Ask forum or to support@iplantcollaborative.org. Thank you for your patience.

Goal

The purpose of this exercise is to gain familiarity with a commonly used procedure for *de novo* whole genome assembly of Illumina reads using the iPlant Discovery Environment (DE). The procedure is based on a publication describing the Assemblathon project, where different methods of assembly were compared.

This procedure will include assembly of paired and unpaired Illumina reads with Soapdenovo2, followed by analysis of the assembly quality for comparison to what was accomplished in one of the Assemblathon procedures that used Soapdenovo1.



Method and Prerequisites

The procedure begins with reads previously trimmed with Scythe to remove extraneous sequence, and Sickle to remove low quality reads and low quality portions of the remaining reads. After assembly with the Soapdenovo 2 App in the DE, the resulting assembly will be analyzed for basic quality statistics, and compared to a reference genome for more in depth analysis of the assembly, specifically for assembly fidelity.

The Assemblathon1 Experiment

The Assemblathon, the first one, was a competition to compare different approaches to genome assembly. For the studies, 2 different genomes were used, but primarily one was tested in the competition: an artificially created genome, the species A genome, which is approximately 12.5 mbp. Synthetic Illumina reads were generated for a diploid version of this genome, and 17 groups from different institutions tried different approaches/assemblers to assemble the genome sequence from the reads provided. Two different groups used Soapdenovo 1, a commonly used whole genome assembler for assembling Illumina sequences. For this tutorial, Soapdenovo 2 will be used for assembly of the same artificial diploid genome, but the exact procedures used to create the final assemblies entered for the Assemblathon. For the tutorial, a recommended approach will be followed in testing different kmer settings, and in this, testing the benefits of using the GapCloser program that is provided by the same group that created Soapdenovo 2.

Test Data

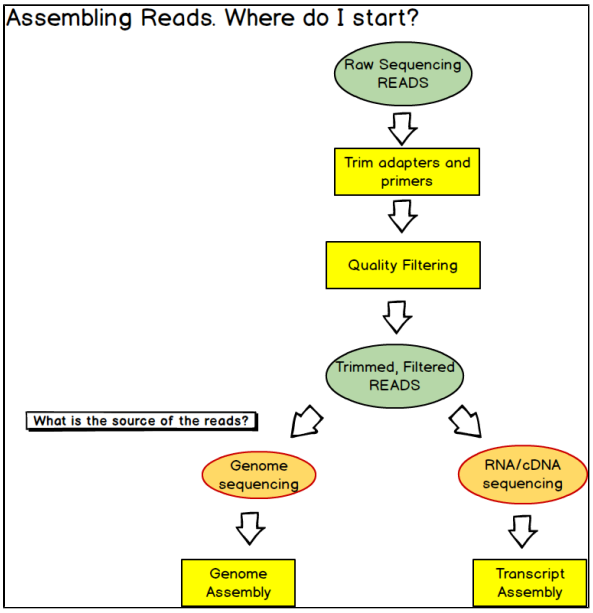
The original Assemblathon1 data is located in the iPlant Data Store at this location: /iplant/home/shared/iplant_assembly_test_data/assemblathon1. The data to be used in this tutorial is available in the Data Store *(Community Data > iplant_training > genome_assembly2 > input_reads).*

This represents the original data from the first assemblathon, trimmed and cleaned up with the applications Scythe and Sickle, as described in a separate tutorial. During the process of trimming, many culled reads left unpaired mates behind, which were moved into a separate single reads file. All the single read files were combined together and renamed to spA_singles.fq.

More information on the Assemblathon is available here: http://assemblathon.org/assemblathon1.

The Assemblathon1 publication, Dent, E. et al., Assemblathon 1: A competitive assessment of de novo short read assembly methods, Genome Res. 2011. 21:2224-2241, is found here: http://genome.cshlp.org/content/21/12/2224.full?sid=74019122-f944-4ccc-bffe-d16fdd0e7d6c.

Procedures



I. Soapdenovo 2

App: **Soapdenovo 2.0.4** (*Public Applications>NGS>Assemblers*).

A de novo, whole genome assembler for Illumina reads created by BGI.

Basic documentation

Time for execution of the tutorial data, approximately 3 hours.

1. Open a Data window by clicking **Data** in the Discovery Environment.
    a. Navigate to the test data (*Community Data>iplant_training>genome_assembly2>input_reads)*. Keep this window off to the side so you can drag and drop the test data into the app window.
    Explanation. General information on the reads was provided with the reads on the Assemblathon1 website (http://korflab.ucdavis.edu/Datasets/Assemblathon/Assemblathon1/README.txt). Typically for real experiments, this information can be provided by the core facility that performs the sequencing. Paired-end reads are commonly provided as f*orward/reverse* reads, and mate pairs as *reverse/forward*. The 3000 bp and 10000 bp insert read files are described as mate pair data. The unpaired data is not used for the scaffolding process, which uses the pairing information to link contigs together.)
2. Click on Apps and search for **Soapdenovo**, select **SOAPdenovo 2.0.4**.
    a. If desired, edit the analysis name and description. It is recommended in your analysis name to indicate the kmer size used (e.g., **Soapdenovo 2.04_analysis1_kmer47**).
3. For App settings, use the following values for test data:

    a. General settings:

    - maximum read size: **100**
    - kmer range: **63mer maximum**
    - kmer: **47** (0.5 x read length, plus 1 is typical, use a little less since many of the reads may be trimmed from their original 100).
    - output prefix name: **Soapdenovo2_spAscsi47**

    b. Paired Reads 1:

    - Insert size: **200**

- read pair orientation: **normal (fr)**
- library1 steps: **3**
- library1 rank for scaffolding: **1**
- library1 file format: **fastq**
- library1 seq1: **spA_200i_40xSCSi.1.fq**
- library1 seq2: **spA_200i_40xSCSi.2.fq**

c. Paired Reads 2:

- Insert size: **300**
- read pair orientation: **normal (fr)**
- library1 steps: **3**
- library1 rank for scaffolding: **2**
- library1 file format: **fastq**
- library1 **seq1: spA_300i_40xSCSi.1.fq**
- library1 **seq2: spA_300i_40xSCSi.2.fq**

d. Paired Reads 3:

- Insert size: **3000**
- read pair orientation: **reverse(rf)**
- library1 steps: **3**
- library1 rank for scaffolding: **3**
- library1 file format: **fastq**
- library1 **seq1: spA_3000i_40xSCSi.1.fq**
- library1 **seq2: spA_3000i_40xSCSi.2.fq**

d. Paired Reads 4:

- Insert size: **10000**
- read pair orientation: **reverse(rf)**
- library1 steps: **3**
- library1 rank for scaffolding: **4**
- library1 file format: **fastq**
- library1 **seq1: spA_10000i_20xSCSi.1.fq**
- library1 **seq2: spA_10000i_20xSCSi.2.fq**

d. Paired Reads 5:

- Insert size: "Leave this setting blank"
- read pair orientation: "Leave this setting blank"
- library1 steps: **1**
- library1 rank for scaffolding: "Leave this setting blank"
- library1 file format: **fastq**
- library1 seq1: **spA_singles.fq**
- library1 seq2: "Leave this setting blank"

e. Options

f. Run Settings

- Maximum Run Time: **4 hours, bigger assembly/genome**

II. Assembly with kmer parameter sweep

4. Click Analyses in the Discovery Environment, select the checkbox next to the Soapdenovo job run above, and then click **Relaunch**.

5. If desired, append **kmer45** to your name and/description. Under the **General Settings** select a **kmer size** of **45** and then click **Launch analysis.**

6. Repeat steps 4 and 5, this time selecting a **kmer size** of **49** (naming/describing your job appropriately)and then launching your analysis.

Note: A best practice is to repeat this same assembly with 2 other kmer settings (e.g., 45 and 49). Creating good assemblies generally requires testing multiple kmer settings with most assemblers in use.

III. Assess assembly.

Time for execution of the tutorial data, approximately 1 hour.

7. Open a Data window by clicking **Data** in the Discovery Environment. Navigate to the test data ***Community Data > iplant_training > genome_assembly2 > assess_assembly > speciesA.diploid.fa***. Keep this window off to the side so you can drag and drop the test data into the app window.

if you are using your own data, additional supported genomes can be found at: (***Community Data > iplantcollaborative > genomeservices > builds > 0.2.1>*** *)*

8. Click on Apps and search for the **Assess Assembly vs whole genome** app (*Public Applications>NGS>Assembly Annotation*). If desired make adjustment to the analysis name, description, etc. (Basic Documentation)

Explanation: Compares an assembly to a genome fasta file and evaluates the fidelity of the assembly.

9. For App **Inputs** use the following values for test data:

- reference.fasta: **speciesA.diploid.fa**
- assembly.fasta: **assembly.fasta** (This is the fasta formatted scaffold file (*.scafSeq) produced by Soapdenovo 2 in steps 1-4. Sample data is available at *Community Data > iplant_training > genome_assembly2 > Soapdenovo2)*
- header: Enter any desired prefix name in the window labeled “header”.

10. Repeat steps 7-9 for the other two Soapdenovo assemblies (kmer size values 45-49 respectively).

11. Compare the results for the different assemblies. Some have larger N50 values for the scaffolds. Some have lower numbers of inserts and deletions. Compare the representation values (the amount of the genome that is accurately reproduced in the assembly).

IV. Additional Assembly: GapCloser

GapCloser is a tool for filling gaps in scaffolds produced with the Soapdenovo 2 assembler. The whole output directory from a successful run of the App Soapdenovo 2.0.4 is ingested and the reads are used to find matches at the internal gap margins of the output scaffold.

Basic documentation

12. Click on Apps and search for **GapCloser**, click on **GapCloser 1.12.0**. If desired make adjustment to the analysis name, description

13. For **Input** enter the entire output directory for a Soapdenovo2 assembly that you want to use with GapCloser. For the test data, (spA genome) enter one of the assemblies from running Soapdenovo2, for example “soapdenovo2_spA47” *Community Data > iplant_training > genome_assembly2 > Soapdenovo2)*

14. For **General Settings** use the following:

- Scaffold name: enter the exact name of the scaffold fasta file found in that assembly directory, for example, “**soapdenovo_spA47.scafSeq**”
- Maximum read length: **100** (the length of the longest reads used in the creating the assembly)
- Overlap setting:
- Maximum Run Time: For the test data, set this to **12 hrs**.

V. Assess assembly on GapCloser outputs

15. Click on Apps and search for the **Assess Assembly vs whole genome** app. (*Public Applications>NGS>Assembly Annotation*). If desired make adjustment to the analysis name, description, etc.

16. For App **Inputs** use the following values for test data:

- reference.fasta: **speciesA.diploid.fa** (Test Data: (*Community Data > iplant_training > genome_assembly2 > assess_assembly > speciesA.diploid.fa)*
- assembly.fasta: **assembly.fasta** (For the GapCloser output from the spA scaffold “soapdenovo_spA47.scafSeq”. The GapCloser output would be automatically named “soapdenovo_spA47.scafSeq_gapcloser.fa”)
- header: Enter any desired prefix name in the window labeled “header”.

Inputs and Outputs

1. Input files:

a. Illumina fastq read files sample:

@assemblathon_20x10000i_1/1

CACAGTTATAACCATTCAAATGTGTTCAGGACTAACTTGTCAGAAAATCAATTTGATAAAGATTCTCAGATCCTTGTTTTTTGGCAAATTGGAACGGGTG

+

dBBecfWfdZTBZ\d`eVf\fc`ecbBaefdab_ebcffBaJe`deffcXfefP`a`beaafTYecddddcb^_ddceT^]lddfe`^dTcVbIdSZfff

@assemblathon_20x10000i_2/1

GGCTCCTCCTCAATTAGACACGAGAGTCACTTTGACCCTCCCTTGTGTGCTTTCCCAATTTCTAACTCTACTTAGTCTTAGCTTTTGCTAAGACCACAAC

+

^BcBTBf]eBededea^bfdBcffb\X`Bacfcfcdccdbdbfff\bBTdQdc_fccddUcdP^eecdd`]ada]`bfce]eYedf\Bdf`f`_fe

@assemblathon_20x10000i_3/1

GTGCATGTCCCCCTGAAGGGTGAGGGTTGGAGTCATGTTGGTAGCCTGTTCATCCTGCAAAGACTTATTGTGAGCAACACCATA
TTTTGTGCTAGTAAAC

+

BY\a]acefT[cecdfadBed`Bd\ZPec\JbcfdcdefTceffa]dfY\Bf^eabf`_c`fdfffbYfBEddBYf]afbf`dfcddffbbfef\ceYcd

@assemblathon_20x10000i_4/1

GAGGTGATATTAGGGCAGGCCAAGGAATTCTGGTTGCATGGTGTGTCTCATGGTGTCACAATCTTAGCATGTAGTGTATTACATT
TTCATCACAAGTAGC

+

efcBV^^c^aXNcbaHf\eeYad\fd\f]NBdfddBeRdddfe[f`ddeZdffffbbd^de\dcbBdbdea`_Bee\ffe`Za^^cdecd^a^b^cd

2. Output files:

a. Scaffold files in fasta format:

>scaffold1 25.6

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAGAGGTAGGTCTCACTTTCACAGTGGTCATGCGATAAGCTTTAC
ATGAGGTCCCTGGA

NNNNNNNNNNNNNCTTTATGAAGTGCCATTGGACACGGGGGCTATAAGCAGTAGTTCAAGATTAAGCATTTTGCATAGAACTTCT
TAGGGAC

>scaffold1 10.7

CCGAGCAAGGGTAGCGGCGCTTACGAGCGCAAGGGCATCGAGCGCAAAGCTCCCGAGCACGGCAGCGCTGGAGAGATCACC
CGAGATCGGCAGCCATGGG

CGCNNNNNNNNNNNNNNNNNNNNNNNNNNCAGGGCGATGCCGGCGACAGGCAGCGCCCAGAATGCAATTCTGCGAGGCAC
GTCAGTAGCCGAGCCAGCG

AAGGACGTCACCCGCNNNNNNNNNNNNNNNNNNNNNNNNNNGTCCCGCAGAAACTCGGCGACGCCTTCTGGCACTGGCACGC
CGTGCACCTTCAGCTCAC

GGAGGGCTCTTTCCTTCGTGATGATCTCGCCCTCTGCGCTCTCGAGGTAGGTCATCATTTTCTTTCCCGCTCCAGTCGATCCGC
AGCCCATTCGGCCGCT

GCGGCCAATCCGGGCGCGCCCCGCGAGAGCTCGTATTCCATCAGCCGCCGAGCGGCCGCCGCGTCGTTCATCACCATGCGCC
GCAGCCTTCGCTCTTGCG

GGTGAAGGCGCGCAGGCCCGGACTGCAACCGCCGAAGGAGGAAGCGGTCTCGGAAACGGCCCGCCAGGATGTCGGCCGCAT
GAAGCAGGCATGAATAATA

GAGAAGGACCGCGCTCCAGATCAGCGCGGTCTGTGCCCCGGCCGGCAGGGCGCGGAACAGAAGCCGCAGGCCCTCTTCCGT
GTCCTGCNNNNNNNNNNC

ATCATCCGTCGTCACCCGTGTCGGCCGGGAAGAGGACCAGCTCGCCATTGACCGGCACGGCGTCGAGATAGGCCTTGAAGAT
GTAGTTCTTCCCCGGCGC

GGCCGGGAACGCCATGCCGAGCGGACGGAAGCGCGTGCGGTTGTCGCTGCCCGTGACCGGCGTCGTGACGCGCATCGGTGC
CGCCGCCATGTCAGGCAGCCGAGGCGGTGCCGCCCGCGCAGCCTGATCCGCCGAGCCGGGGACGAACACGATCAGCTCGCC
ATTGATCGGGTTGGCGTCGAGCCGGATCTTGAAGAGGTAGTTCTGCCCCGGCCGGGAGGGGAAAGCGACGCCGACCGACCGG
AAACGGGTCTGGTCCTCGCCTGCCTTCACAAGTGTGGTCACGTTCAACACGTCAGC

CATGAGCTGGCTCCTTCAACTCGGGAATGGGG

>scaffold2 8.2

GACGCCGCTGAAGTCGTAGACCATGTCATTGCAGGTCAGGGTGTCCCCCTCGGCCCGGAAGACGGTCAGCTCATACATCCGCC
GAAGCGGGACCATGCGA

ATGCGCATNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNACCTGCGTGCCTCCGTAGGACGTCGGAGCCA
GCAGCATGTAAGCGGCGA

CGTTCGAGGCCCCCGGCCGGCAGAGGAGCGAGATCACCGAACCGCCCCGGCAGCCGCCGGTCACGCANNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNCGGAAGCGTCACCGTCTGCCATTCGGTGCGATAGAGCGAGCCCTGCGCGATGGTCGGACCCGGA
ACCCCGGTCTGGACCAGGCGGCAAAGCTGTGTCCCAT

CCGCAAAGCGCACATATTCGGCGCCGGCCGTCTCGCCCTTTTCGATGATGCCGCCGCGGGGGAAGCCGGAGGACCAGCTGAC
AGCGCCCACGATGTTGCG

CTGGTCGTAGGTCAGGAACCAGTCGCCCCAGGTCGTATCCTTCTGCCGGCACCAGCGCCCGGTATCGGTGGCCGTCTGCGGA
TAGGCGATCTGGATGGCG

CGCNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNGACCTGATAGA AGCCCGTCGTCCCGATCTG

GTCCGCATCGTTGCCCGGGATCGGGCGCGCGGTGCCGCCGAGGCCGTAGTCCCCGACCTTCAGGAGGCGGCCGGGCGTGGT GTCGAGATCGCTCTGGGTC

ACGGCGGTGCCGGAGAGAAGCCCCTGCAGCTCCATGCCGGACGGCGTGAACCGCGCCCGTTCCGTCCCCTGGCAGGTGACG CCGATCTCGTTCTCGGCCG

CGAGGAAGAAGCCGGTG

b. Scaffold fasta file after GapCloser:

>scaffold3 7.4

GGCCATGGCCGAGAGCCTCGCCGTGGCGGGGCTGCTGCAGAACCTCATCGGCCATCTGACGCCCCAGGGCGTCGAGATCGT GGGCGGCGGCACCCGGCTC

CGCGCGCTGCAGCGCCTCGCGGCCGAGGGCTGGAGCCGGCACCCGGATCTCATTCCGATAGATCCGGTACCGGTGAAGGTG ACGGCCGACCTGCAGGAGG

CGGTGGCCTGGGCGGGGACCGAGAACAGCGCGCGCTCGGCGCTGCATCCGGCCGACGAGGTCCGCGCCTATGCCGCGATGC GCGACCGCGGCGCGAGCCTGTCGCGGATCGCGCGCAGCTTTGCGCGCTCCGAGGCNNNNNNNNNNNNNNNNNNNNNNNNNCT GCCGGCCGAGGCGCTGGCGGCGCTGCGGGCGAACGAGATCTCGCTCGAGATGGCGAAGGCGCTGACGCTCGCGCCGAGCG GCGAGCGCTGCCTCGAGGTGCTGGGATCGGTGCGCGGCCGCGACATGCGGCCCGAGCAGGTCCGGCGCGAGCTCACGCCC GGCACCGTGCCCTCGACCGACCGCCGCGTCCTCTTCGTGGGGCTGGAGGCTTACCTTGCTGCCGGCGGAACCTCTCAGCGCG ATCTCTTCGCCGACCGGACGCTGCTGGAGGACGAGGCGCTCCTCGACCGGCTCTTCGCCGAGAAGGGCGCGGCCGAGGCCG AGCGGATCCGCGCGGAAGAGGGCTGGGAATGGGCGACGTGGGTGCCGGAGGAATATGTCTCCTGGACCGTCACGCAGAAGC TG

>scaffold12 3.0

CTCCTCGAGCCCGCGCAGCTCGAGCGCCCCGTTCACGAGCCGGGGGCCGAGGCCCGCGAGGAAGGCCGCGCCTTTCACGGG ATCGACCAGCAGCGGCGTGCCGAACACCCGGCCCGCGATCATCGGATAGTTCATGGCTCTGCCCTCCTGCCGGCGTCGCGGT CCGCCGCGCGTCCGTCGTCGTCCTCATCGTCCTCCTC

GTCGGCGCGCGCCCCGTCTTGCGTCGCAGGCGCGGANCCTGCGTCGGAGGCGCGGAGGAGCCCTGCCCGCTGCGGCGGAA GTCGAGCCCGAACCGCCCCTCACGCTTGCGGTCGGCGGCGATCTGGCGGTCGACCTCCTCGGCGTCGTAGCCGCGCTCTCCG ATCGCCTGCGACCGGCTCTTGAGGCCCGCGCCGATCTCCTCGATCTCGGCGCGGATGTCCTTCAGCGGATCGACCCATTGCCA CTTCGGCGGCAGCCACTCGCAGGCGAGATAGTCCCGCCGCCGCC

1. Contig fasta file (not assembled in scaffolds):

>12629 length 638 cvg_9.1_tip_1

GCCGGAGCGGGGCTTATGCCGGGCGCCGCAGTGCCATGACGCAGCGGACCGGGATGTGGAAGTAGCGCCCGATCTGGGCGT CCGTGAGGCCGAGATCGGT

GAGCGTATCGATCACCCGATCGGGCGAGGGCGCCATCAGCACCTGCGCGCAGAGGCAGGCCATATGGCCGCGGGCGTCCAG TTCATCCTGACCGGGGCAG

AGGCCGGTCGCCTTCCAGATGCAGCTCGTTTCATCGGGCTCTGCCGAACGGACGGCCTCGTCACGGGTCGATGGGGGCGGAG CGGTCCTGCGGCTGCGGA

ACAGGGCCCGGCGCCGTTCGGTCGTTTCGGTTATCATGGGGCAGTCCTCCTCTTCCTGCACCATTGCCGCCTGAGCGACAGGC TGAAGTCTCGCCGTTAC

CTGACCGAAGTCAATCTTCATCTGCGTGACGGCCGGATGGCTATCGCACAGGCTTGTAAACTTTTTCCTAACAAGCGGCTGTTTT ACCGTCCTGTGCCGGCCAGCCTCCGCGCGGGGATCCGGGCAGGAAATCTGCCGGCACCGGCGCGGCGCCCCGAGACCGATG CGCCAGCGACGGTTCACCCTCGGGATGTGCCGGCGCGCTGATTCTCGCCTCGGGTTGCGCGGACGGCCGGCC

>12631 length 638 cvg_6.0_tip_1

CAGCGGGGCGAGGGCACGCCCAGCCCGATGGTGCAGGTGGTCTCCTCGGTCACGCCGGCCGCGGTGCCGGCCGCGAGGAT CCGGTCGCGCGCCCGCGTCGTGTCGGCCGAACAGGCGTCGAAGCGCGCACCCTCCTCGTCCATCACGAAGCGCAGCGTGAA GGGCGTGAGCACGGGCCGCGGCGCCGAGATGTTCATGACGAGCTCCAGCCCCTCTGGCGCGAGCCGCAAAAGCTGCCCCTC GAGCTTGCGCTTCTCGTCCTCGCTGTTCGAGATGGCCGTGATCGAGACCCGGTCGGCA

TCGACCGAGATCTTGGATTGCGGCAGGAGGCGCAGGGCCTCGAGCCCGAAGGCGAGCGCCGAGGACCAGCCCTCGGGCGTC GGGAAATCGGCAGTCTCCA

GCATGTCGGCCACCTGAAGCCCGGTGCCGAGGCCGCGCGCGGCATTGGCCAGATCCTCGGTGCCGCCCGCCTCGGGCAGCA GGCCGATCAGCGACACGCC

GTCGTCGTTGCGCAGCATCTCGACCGAGAAGCGCGGCGCCTCGATCTCGCGCACGGGCGTCACCTCGAGCGCGTCGCGCAG CCGCGTGGCATCGACGGCC

CCCCCCGCGAGGTTCACCGCGCGGAAGCGCGTGGCCTC

Application	Output	Output Type	Content
Soapdenovo 2	*.scafSeq	fasta format text file	scaffold fasta file, which represents the final output for the assembly sequences (includes large gaps filled with N's)
Soapdenovo 2	*.contig	fasta format text file	contig fasta file, the final assembly of continuous sequence (only small gaps)
Soapdenovo 2	*.scafStatistics	text	text formatted list of statistics about the scaffold sequences including the N50 value
Assess assembly	*.report	text	text formatted list of statistics about the scaffold sequences (or any input fasta file), including N50, but also comparisons to a reference genome (e.g. representation value)
GapCloser	*_gapcloser.fa	fasta format text file	scaffold fasta file, which represents the final output for the assembly sequences from GapCloser