DEWE v1.1

USER MANUAL

Table of contents

1. Introduction

Transcriptomic profiling aims to identify and quantify all transcripts present within a cell type or tissue at a particular state, providing information on which genes are being expressed in precise experimental settings, differentiation or disease condition. This technique is thus essential for discerning how changes in gene expression relate to functional changes in the organism or tissue and is the only "omic" approach capable to shed insights into transcriptional regulation, signalling pathways and gene network organization [1]. Traditional transcriptomic approaches were based on microarrays RNA-DNA hybridization, but high-throughput sequencing of mRNA (also called RNA-seq) is a powerful tool which offers many advantages over hybridization-based studies. Deep sequencing allows theoretically for identification and quantification of all mRNA presents within a cell type at a specific condition, including non-coding RNAs and small RNAs in a single experiment and with high accuracy. Increasing sequencing depths in new platforms have even made possible to perform dual-RNA-Seq, performing simultaneously transcriptomic studies in interacting organisms allowing for instance the characterization of pathogen- host interactions within a single experiment [2]. In addition, RNA-Seq can identify transcripts "*de novo*" as it is not dependent on previous probes design and synthesis [3]. Therefore, RNA-seq is becoming a standard for transcriptomic studies. The many advantages of RNA-seq are partly possible due to the generation of an enormous number of raw sequencing reads, typically tens of millions for a standard experiment, which allow capturing even low abundant transcripts. Consequently, RNA-Seq data analysis requires the utilization of specific software designed to handle with the vast amount of data generated in these experiments, whose utilization can be challenging for the non-familiarized user, due to the volume and complexity of data produced and the absence of Graphical User Interfaces (GUI).

As the popularity of RNA-seq grew in the last years, a high number of data analysis methods and software tools have been developed for different tasks [4]. The main stages for a differential expression workflow are: (*i*) reads alignment, for which Bowtie2 [5] or HISAT [6], among others, are available; (*ii*) transcript assembly and quantification, for which StringTie [7], Cufflinks [8] or iReckon [9], among others, are available; and (*iii*) the differential expression analysis itself, for which tools such as Ballgown [10], edgeR [11], DESeq [12], baySeq [13], or Cuffdiff [8], among others [14], are available. Also, complete workflows that combine these tools have been published and discussed [15][16][17]. However, many of these tools are difficult to install, configure or use for end-users such as life-scientists without medium to advanced bioinformatics skills. For these reasons, a great variety of interfaces for RNA-Seq analysis have also been developed, trying to facilitate the work of such users [18].

Examples of such interfaces are easyRNASeq [19], Galaxy for RNa-Seq [20], RNASeqGUI [21], RobiNA [22], RSeqFlow [23], or SePIA [24]. These interfaces have been reviewed by Poplawski et al. , who encountered several technical difficulties in installing, configuring and using them [18]. They also pointed out that both the limited flexibility in analysis steps and such unexpected technical difficulties might shift the balance in favor of established command-line-based protocols. Container-based technologies such as Docker (https://docker.com/) have been developed to overcome these challenges by automating the deployment of applications within the so-called software containers. A software container offers an isolated environment for the installation and execution of a specific software, without affecting other parts of the system. Different groups have proposed the use of Docker containers to solve bioinformatics problems [25].

In this scenario, there is room for developing a new RNA-Seq software tool that overcomes this issues.Thus, we present **DEWE, a tool to perform complete DE analysis workflows in eukaryotic RNA-Seq data** (comparing two conditions with at least two samples each), allowing also users to individually perform each supported step, including raw reads quality control and filtering. DEWE can be easily installed by any life-scientist without advanced bioinformatics skills, requiring minimal or zero configuration. Thanks to its user friendly interface and the comprehensive documentation provided, users may be familiarized with the interface in a short period of time.

1.1. The SING research group

The SING research group (http://sing-group.org/) has been developed many Bioinformatics applications since the last 12 years.

Other related developments:

- MAHMI database: a comprehensive MetaHit-based resource for the study of the mechanism of action of the human microbiota (DOI: 10.1093/database/baw157).
- P4P: a peptidome-based strain-level genome comparison web tool)DOI: 10.1093/nar/gkx389).
- BlasterJS: a BioJS component for interactive visualisation of BLAST alignments results.
- RUbioSeq+: a multiplatform application that executes parallelized pipelines to analyse next-generation sequencing data (DOI: 10.1016/j.cmpb.2016.10.008).

1.2. Funding

This work was supported by the Spanish “Programa Estatal de Investigación, Desarrollo e Inovación Orientada a los Retos de la Sociedad” (grant AGL2013-44039R); the Asociación Española Contra el Cancer ("Obtención de péptidos bioactivos contra el Cáncer Colo-Rectal a partir de secuencias genéticas de microbiomas intestinales", grant PS-2016). This study was also supported by the Portuguese Foundation for Science and Technology (FCT) under the scope of the strategic funding of UID/BIO/04469/2013 unit and COMPETE 2020 (POCI-01-0145-FEDER-006684); and the INOU16-05 project from the University of Vigo. SING group thanks CITI (Centro de Investigación, Transferencia e Innovación) from University of Vigo for hosting its IT infrastructure.

1.3 Third-party software

Table 1 shows the specific third-party tools and versions that DEWE uses for each step of the analysis.

Table 1. Third-party software used in DEWE.

Name	Website	Version
FastQC	https://www.bioinformatics.babraham.ac.uk/projects/fastqc/	0.11.5
Trimmomatic	http://www.usadellab.org/cms/?page=trimmomatic	0.36.0
Bowtie2	http://bowtie-bio.sourceforge.net/bowtie2/index.shtml	2.3.0
HISAT2	https://ccb.jhu.edu/software/hisat2/	2.0.5
SAMtools	http://samtools.sourceforge.net/	1.1.3
StringTie	https://ccb.jhu.edu/software/stringtie/	1.3.1c
HTSeq	https://htseq.readthedocs.io/en/release_0.9.1/	0.6.1
R	https://www.r-project.org/	3.4.4
Ballgown	https://bioconductor.org/packages/release/bioc/html/ballgown.html	2.6.0
EdgeR	https://bioconductor.org/packages/release/bioc/html/edgeR.html	3.16.5

2. Installation

DEWE is available under two different installation methods: through a Docker container or through the use of a VirtualBox Machine.

The official Docker image of DEWE is available at our DockerHub repository: https://hub.docker.com/r/singgroup/dewe/. This image contains a installation of DEWE with all the necessary dependencies already installed and configured. The main advantages of using Docker is that the end-user does not need to install anything but Docker. Moreover, the provided installers install all the necessary components to run DEWE through the Docker image.

In addition, a VirtualBox Virtual Machine with DEWE and all its dependencies is available for download.

2.1 Docker installers

The Docker installers are available for the following operating systems:

- Windows 7 64 bits or higher.
- Linux 64 bits with 3.10 kernel minimum.
- Mac OS X 10.8 "Mountain Lion" or newer.

For this type of installation, a *shared folder* must be defined between the Docker container and the host machine. In this folder will be stored in both the files of each case study and their results. Any other folder inside the host machine other than the *shared folder* will not be accessible through DEWE.

2.1.1 Windows Installer

WARNING: the is a beta and therefore a non-stable version, errors may occur during execution.

2.1.1.1. Prerequisites

To install DEWE, the machine must have installed a 64-bit operating system running Windows 7 or higher. Additionally, the virtualization option should be enabled, following the manufacturer's instructions.

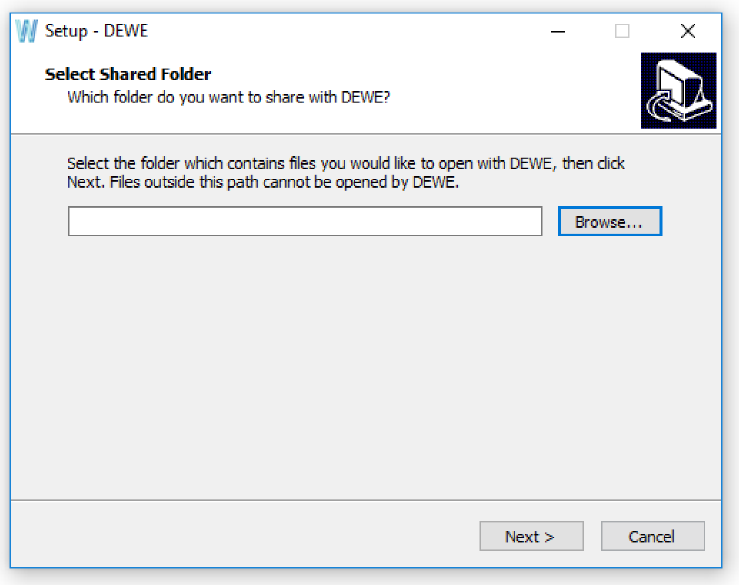
Depending on system configuration, the antivirus software may need to be paused to install DEWE correctly.

The DEWE installer for Windows is available at the following link http://static.sing-group.org/software/DEWE/installers/1.1/DEWE-docker-1.1.exe.

2.1.1.2. Installation

After executing the installer, the *shared folder* has to be specified.

This folder will be accessible from DEWE. Other folders outside this one will not be accessible within DEWE.



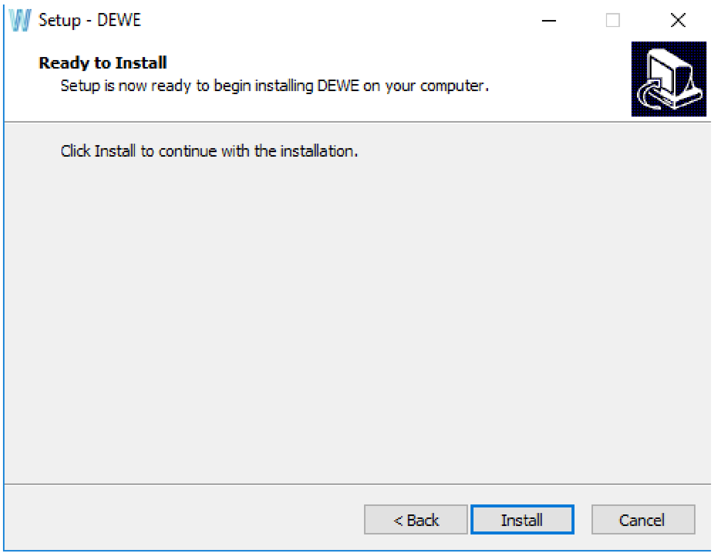
Setup - DEWE

Select Shared Folder

Which folder do you want to share with DEWE?

Select the folder which contains files you would like to open with DEWE, then click Next. Files outside this path cannot be opened by DEWE.

The next step is click to proceed with the installation.



Setup - DEWE

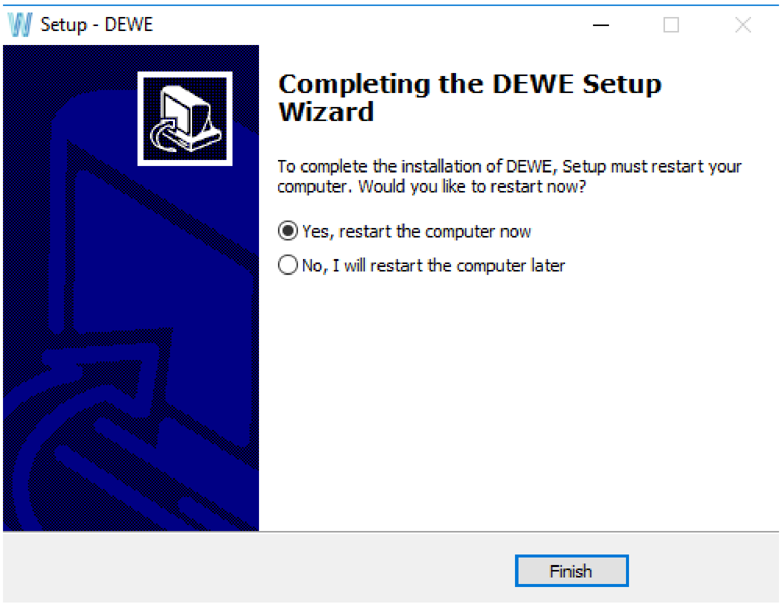
Ready to Install

Setup is now ready to begin installing DEWE on your computer.

Click Install to continue with the installation.

Setup will install DEWE and all its dependencies. Installation can take several minutes to complete. Depending on the configuration of the Operating System, it may be necessary to accept some alerts and the installation of additional drivers.

Once the setup has completed, the computer will have installed DEWE as well as a fully functional VirtualBox, Docker and Xpra client applications.



Completing the DEWE Setup Wizard

To complete the installation of DEWE, Setup must restart your computer. Would you like to restart now?

● Yes, restart the computer now

○ No, I will restart the computer later

The last step is finish the installation and restart the computer.

2.1.1.3 First run

DEWE is now ready to start working. The tool can be launched using the DEWE shortcut at the desktop or from the Start Menu folder.

The first time, DEWE will take a while to start, because it has to download the last available version of the DEWE docker image (~600MB) and execute it.

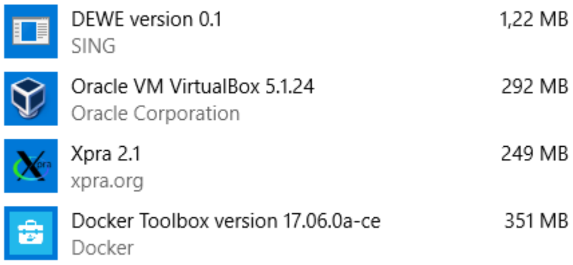
2.1.1.4 DEWE Viewer

As we have seen in the previous section, to execute DEWE in Windows the user has to execute the *DEWE* shortcut.

As the start of the tool consumes certain amount of time, at the time of closing DEWE the user has two options. On the one hand, let it run in the background, and on the other hand close it completely. If it has been decided to let it run in the background, the user can reduce considerably the time to restart the application using the *DEWE Viewer* shortcut. This shortcut will do, instead of rebooting DEWE, move the tool from second to foreground. If, on the contrary, the tool has been completely closed, the *DEWE* shortcut must be used to restart the tool.

2.1.1.5 Uninstallation

DEWE can be uninstalled using *Add/Remove Programs* from Control Panel. DEWE and the main dependencies can be uninstalled separately, i.e. VirtualBox, Xpra and Docker Toolbox.



DEWE version 0.1 SING	1,22 MB
Oracle VM VirtualBox 5.1.24 Oracle Corporation	292 MB
Xpra 2.1 xpra.org	249 MB
Docker Toolbox version 17.06.0a-ce Docker	351 MB

2.1.2 Linux Installer

2.1.2.1. Prerequisites

DEWE requires a 64-bit installation regardless of the computer Linux version. Additionally, the computer kernel must be 3.10 minimum. To check the current kernel version, open a terminal and use **uname -r** to display your kernel version.

The DEWE installer for Linux is available at the following link http://static.sing-group.org/software/DEWE/installers/1.1/linux/install-dewe.sh.

Also, the DEWE uninstaller is available at http://static.sing-group.org/software/DEWE/installers/1.1/linux/uninstall-dewe.sh.

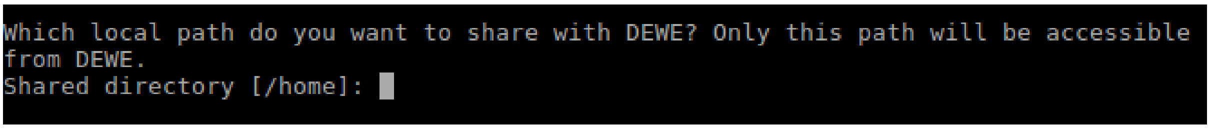
2.1.2.2. Installation/DEWE/installers/1.1/linux/dewe-docker.sh

On the installer download folder open a terminal and execute this command as root:

```
$ sh ./install-dewe.sh
```

To be able to share files with DEWE, a path must be selected.

The installer will ask for a path, by default /home will be selected as shared directory.



```
Which local path do you want to share with DEWE? Only this path will be accessible
from DEWE.
Shared directory [/home]:
```

Setup will install DEWE and all its dependencies. It can take several minutes to complete.

Once the setup has completed, the computer will not only have DEWE installed, but a fully functional Docker and Xpra client applications.

2.1.2.3. First run

DEWE is now ready to start working. You can open it from the DEWE icon on the Start Menu or typing on terminal:

```
$ dewe
```

The first time it will take a while to start because it has to download the last version available of the DEWE docker image (~600MB) and execute it.

2.1.2.4. Uninstallation

On the uninstaller download folder open a terminal and execute this command as root:

```
$ sh ./uninstall-dewe.sh
```

The uninstaller will prompt to uninstall the dependencies of Docker and XPRA before starting the DEWE uninstallation process.

Once the uninstaller execution is complete, DEWE (and its dependencies if selected) have been removed from the computer.

2.1.3 Mac OS X Installer

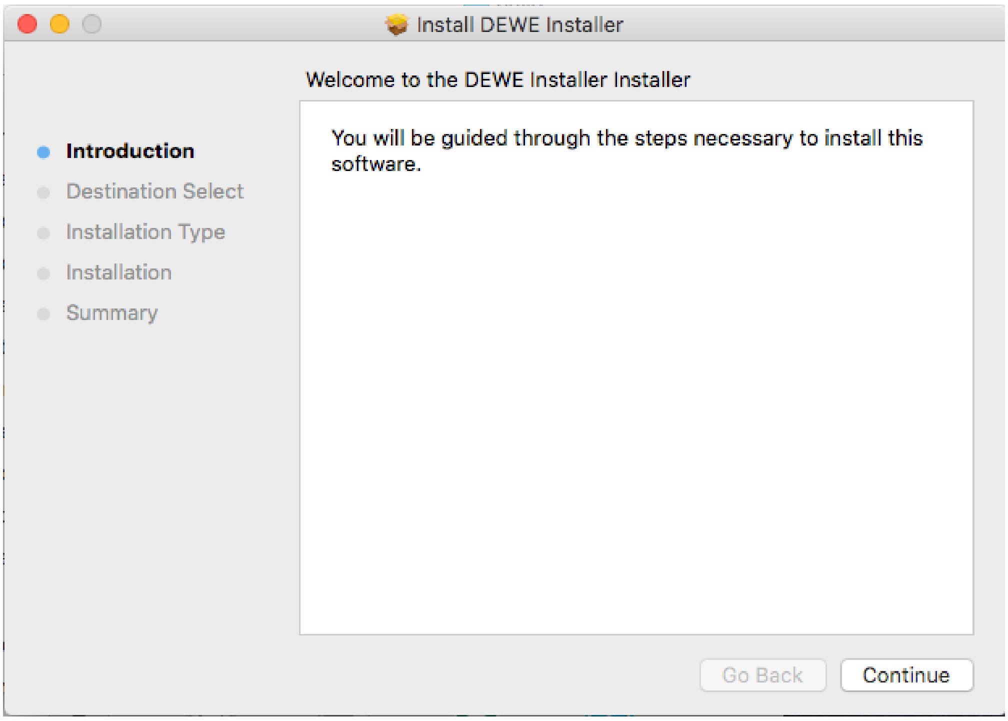
WARNING: this is a beta and therefore a non-stable version, errors may occur during execution.

2.1.3.1. Prerequisites

To install DEWE, the MAC machine must be running OS X 10.8 "Mountain Lion" or newer. The DEWE installer for Mac OS is available at the following link http://static.sing-group.org/software/DEWE/installers/1.1/DEWE-docker-1.1.pkg.zip.

2.1.3.2. Installation

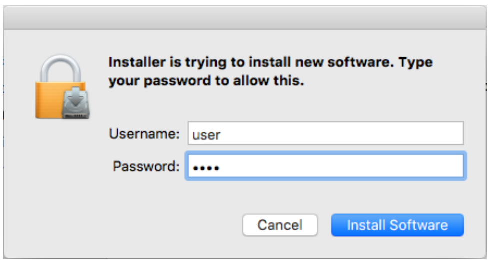
Execute the installer and accept the terms of the installation. Once the setup has completed, you will not only have DEWE installed, but a fully functional VirtualBox, Docker and Xpra client applications.



Welcome to the DEWE Installer Installer

You will be guided through the steps necessary to install this software.

To proceed with the installation, you will need your admin password when prompted.



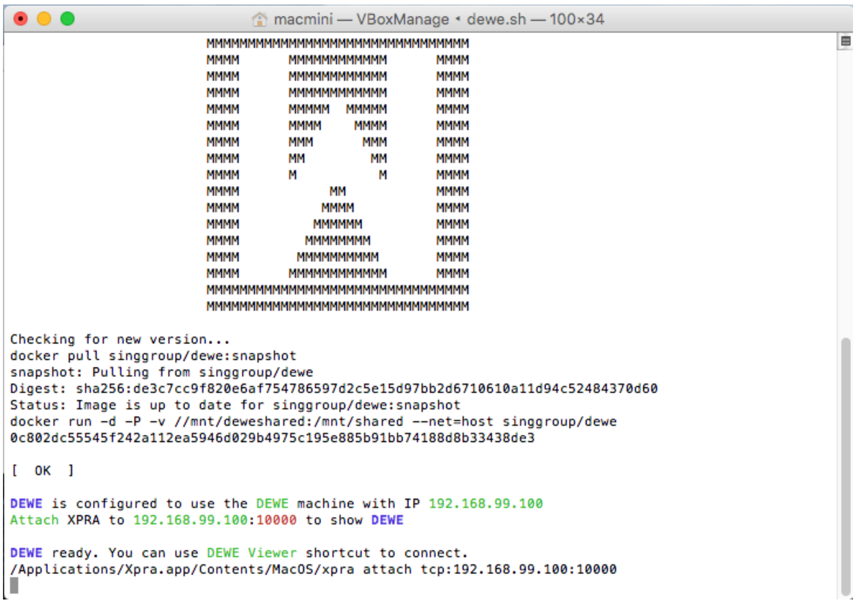
Installer is trying to install new software. Type your password to allow this.

Username: user

Password: ••••

2.1.3.3. First run

By opening the DEWE application from the Applications folder, Terminal window will be opened where you can see the progress of the configuration and the download of the Docker image (~600MB). The download of the Docker image is only required the first time you run DEWE and may take a while, depending on your network connection.



```
                        MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
                        MMMM      MMMMMMMMMMMM      MMMM
                        MMMM      MMMMMMMMMMMM      MMMM
                        MMMM      MMMMMMMMMMMM      MMMM
                        MMMM      MMMMM  MMMMM      MMMM
                        MMMM      MMMM    MMMM      MMMM
                        MMMM      MMM      MMM      MMMM
                        MMMM      MM        MM      MMMM
                        MMMM      M          M      MMMM
                        MMMM           MM           MMMM
                        MMMM          MMMM          MMMM
                        MMMM         MMMMMM         MMMM
                        MMMM        MMMMMMMM        MMMM
                        MMMM       MMMMMMMMMM       MMMM
                        MMMM      MMMMMMMMMMMM      MMMM
                        MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
                        MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM

Checking for new version...
docker pull singgroup/dewe:snapshot
snapshot: Pulling from singgroup/dewe
Digest: sha256:de3c7cc9f820e6af754786597d2c5e15d97bb2d6710610a11d94c52484370d60
Status: Image is up to date for singgroup/dewe:snapshot
docker run -d -P -v //mnt/deweshared:/mnt/shared --net=host singgroup/dewe
0c802dc55545f242a112ea5946d029b4975c195e885b91bb74188d8b33438de3

[  OK  ]

DEWE is configured to use the DEWE machine with IP 192.168.99.100
Attach XPRA to 192.168.99.100:10000 to show DEWE

DEWE ready. You can use DEWE Viewer shortcut to connect.
/Applications/Xpra.app/Contents/MacOS/xpra attach tcp:192.168.99.100:10000
```

After this process, the application will be started and the main window is showed. All files under the home directory are available to DEWE.

2.1.3.4. Uninstallation

You can uninstall DEWE and the main dependencies separately: VirtualBox, Xpra and Docker Toolbox. To uninstall them just drag the app from the Applications folder to the Trash.

2.1.4 Docker installers FAQ

This section contains the Frequently Asked Questions about DEWE and several known problems that may occur when using the application:

2.1.4.1. Windows installer

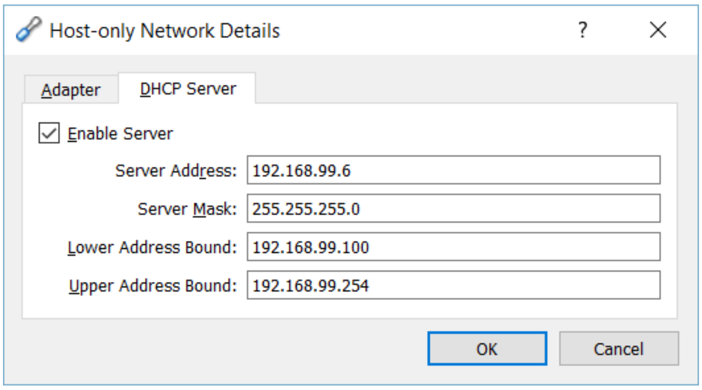
2.1.4.1.1. Error checking TLS connection

Error checking TLS connection: Error getting driver URL: Something went wrong running an SSH command!

This error may happen the first time DEWE is started and is related to the network configuration of VirtualBox. To fix it, it is necessary to check the network configuration in VirtualBox.

Open VirtualBox and go to menu File > Preferences. Then select Network section and go to Host-only Networks tab. Select "VirtualBox Host-Only Ethernet Adapter #2" and edit its configuration. On the DHCP Server tab, DHCP service must be enabled with the following configuration:

- Server Address: 192.168.99.6
- Server Mask: 255.255.255.0
- Lower Address Bound: 192.168.99.100
- Upper Address Bound: 192.168.99.254



☑ Enable Server

Server Address: 192.168.99.6

Server Mask: 255.255.255.0

Lower Address Bound: 192.168.99.100

Upper Address Bound: 192.168.99.254

2.1.4.1.2 Errors occurred. See the logfile xpra.exe.log for details

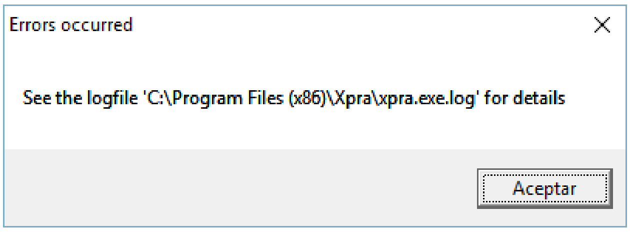
Errors occurred. See the logfile xpra.exe.log for details

This error may occur because there are some permission problems in the installation path of Xpra on Windows 10.

The current workaround is to use the shortcut *DEWE Viewer* available in the Start Menu

2.1.4.1.3 Errors occurred

Errors occurred: See the logfile 'C:\Program Files (x86)\Xpra\xpra.exe.log' for details


Errors occurred

See the logfile 'C:\Program Files (x86)\Xpra\xpra.exe.log' for details

This error may happen every time DEWE is started in Windows and it is related to a permissions problem when starting the Xpra client.

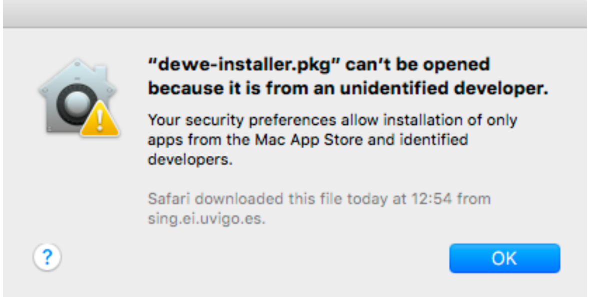
The current workaround is to use the DEWE Viewer shortcut available in the folder DEWE under All programs.

2.1.4.2. MAC OS X installer

2.1.4.2.1 Installer can't be opened because it is from an unidentified developer

"dewe-installer.pkg can't be opened because it is from an unidentified developer.

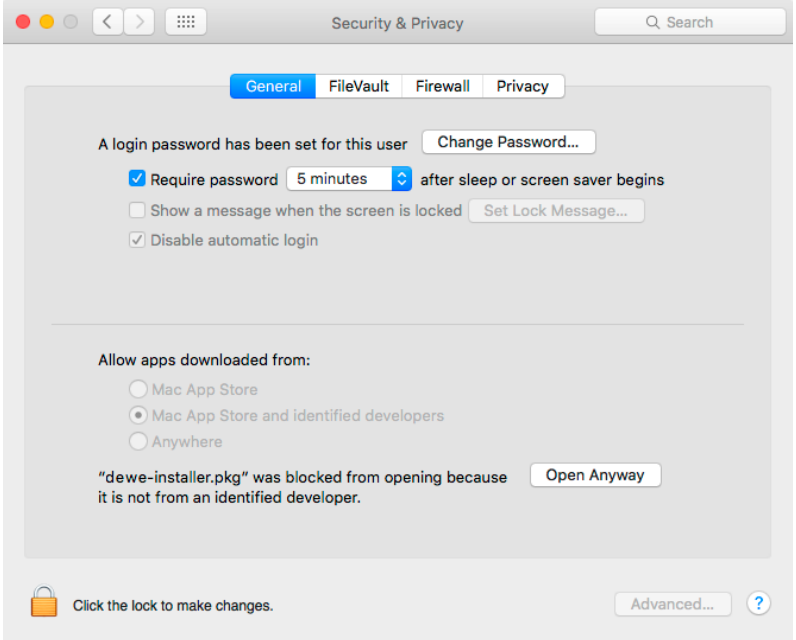
Your security preferences allow installation of only apps from the Mac App Store and identified developers.


"dewe-installer.pkg" can't be opened because it is from an unidentified developer.

Your security preferences allow installation of only apps from the Mac App Store and identified developers.

Safari downloaded this file today at 12:54 from sing.ei.uvigo.es.

In this case, you need to allow the execution of the installer in the system settings: "Security & Privacy". There, click the button "Open Anyway" to launch the installer, or allow apps downloaded from Anywhere.



A login password has been set for this user

☑ Require password 5 minutes after sleep or screen saver begins

□ Show a message when the screen is locked

☑ Disable automatic login

Allow apps downloaded from:

○ Mac App Store

● Mac App Store and identified developers

○ Anywhere

"dewe-installer.pkg" was blocked from opening because it is not from an identified developer.

Click the lock to make changes.

2.1.4.2.2 Some characters are missing in Mac version

In the Mac OSX version of DEWE there are some incompatibilities with several keyboard layouts like ES-ISO. Some keys may print spaces instead of the character associated to the key. The current workaround is to use another keyboard layout like ES or EN-US.

2.1.4.3. Linux installer

2.1.4.3.1 Client is newer than server

```
docker: Error response from daemon: client is newer than server (client API
version: 1.22, server API version: 1.21).
See 'docker run --help'.
```

This error may occur when updating DEWE. In this case, you need to upgrade your docker image using the following command:

$ *docker-machine upgrade*

Once it finishes the upgrade, you need to restart DEWE virtual machine from VirtualBox or reboot your system.

2.2 Virtual machine

2.2.1. Prerequisites

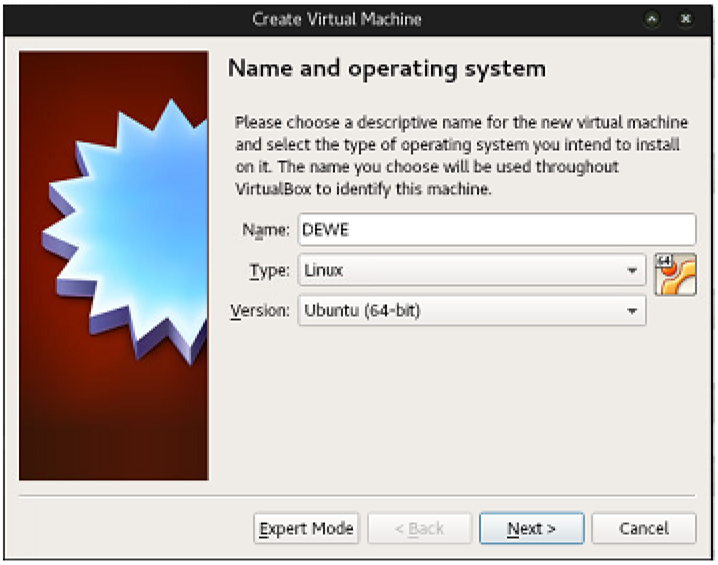
The installation of the VirtualBox is required to use the DEWE Virtual Machine. The installation files and instructions for this program can be found at the official webpage (https://www.virtualbox.org/).

The DEWE Virtual Machine is available at the following link http://static.sing-group.org/software/DEWE/installers/1.1/DEWE-VM-1.1.vdi.

2.2.3. Installation

Decompress the downloaded file (you will need a ZIP decompressor). Once decompressed it should be a file called *DEWE-VM.vdi* with is the virtual machine hard disk. Now the DEWE Virtual Machine is ready to use.

Open VirtualBox and click on the New button. Give a name to your virtual disk and under Operating system choose Linux (the Ubuntu version will be automatically selected). Click on the *Next* button.



Create Virtual Machine

Name and operating system

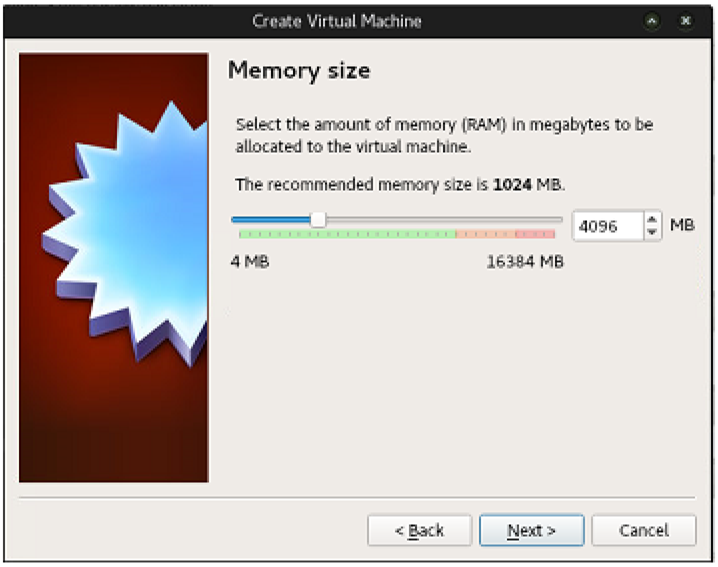
Please choose a descriptive name for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

Name: DEWE

Type: Linux

Version: Ubuntu (64-bit)

Choose the amount of RAM memory to be allocated to the virtual disk. Choose at least **4GB**. Click on the *Next* button.



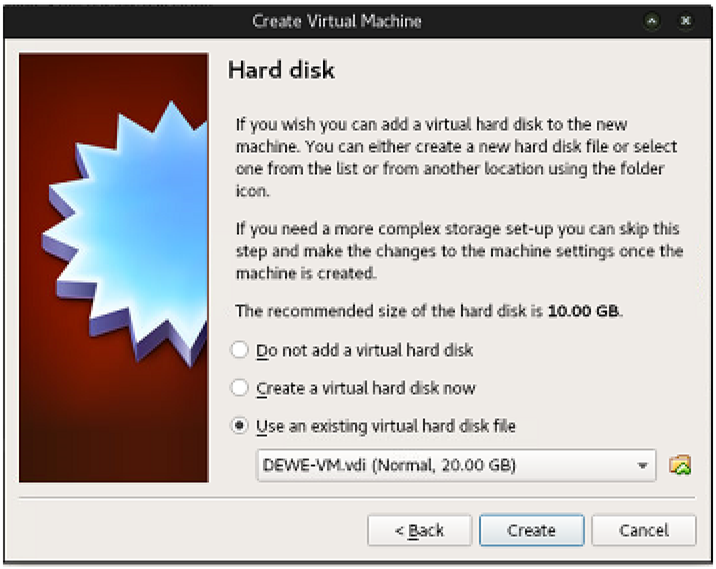
Memory size

Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **1024** MB.

4096 MB

Choose the *Use an existing virtual hard drive file* option and select the location of the *DEWE-VM.vdi* file you extracted (by clicking on the icon that looks like a folder). Click on the *Open, Next,* and then *Create* buttons.



Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select one from the list or from another location using the folder icon.

If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

The recommended size of the hard disk is **10.00 GB**.

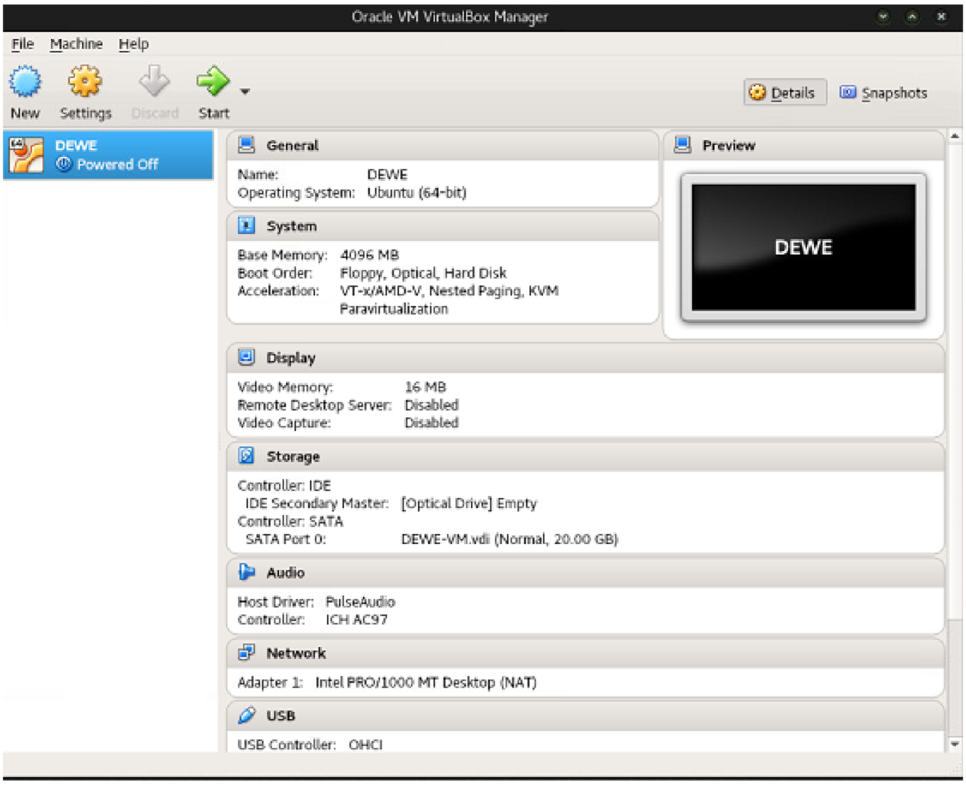
○ Do not add a virtual hard disk

○ Create a virtual hard disk now

● Use an existing virtual hard disk file

DEWE-VM.vdi (Normal, 20.00 GB)

After creating the virtual machine, a shared folder must be selected.



General
Name: DEWE
Operating System: Ubuntu (64-bit)

System
Base Memory: 4096 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, KVM Paravirtualization

Display
Video Memory: 16 MB
Remote Desktop Server: Disabled
Video Capture: Disabled

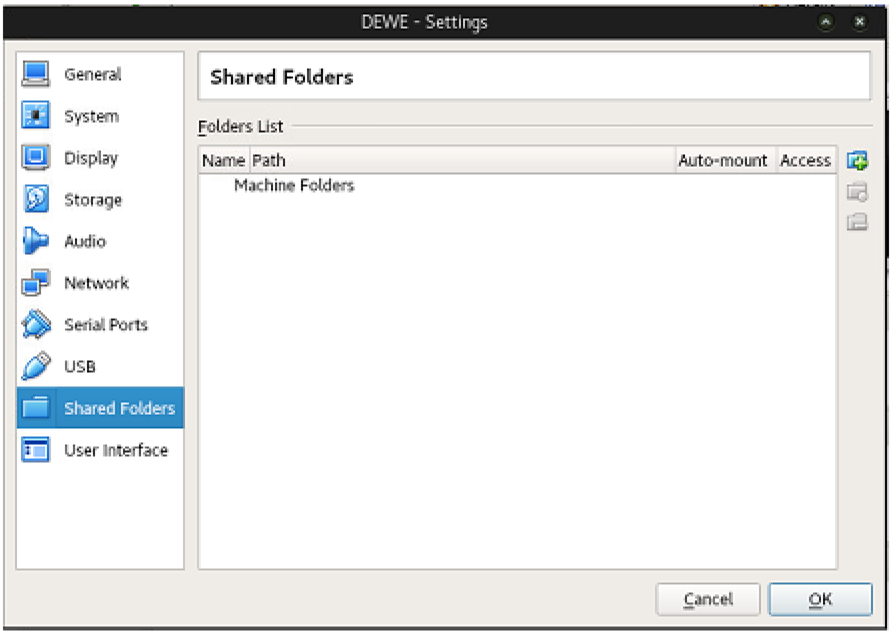
Storage
Controller: IDE
IDE Secondary Master: [Optical Drive] Empty
Controller: SATA
SATA Port 0: DEWE-VM.vdi (Normal, 20.00 GB)

Audio
Host Driver: PulseAudio
Controller: ICH AC97

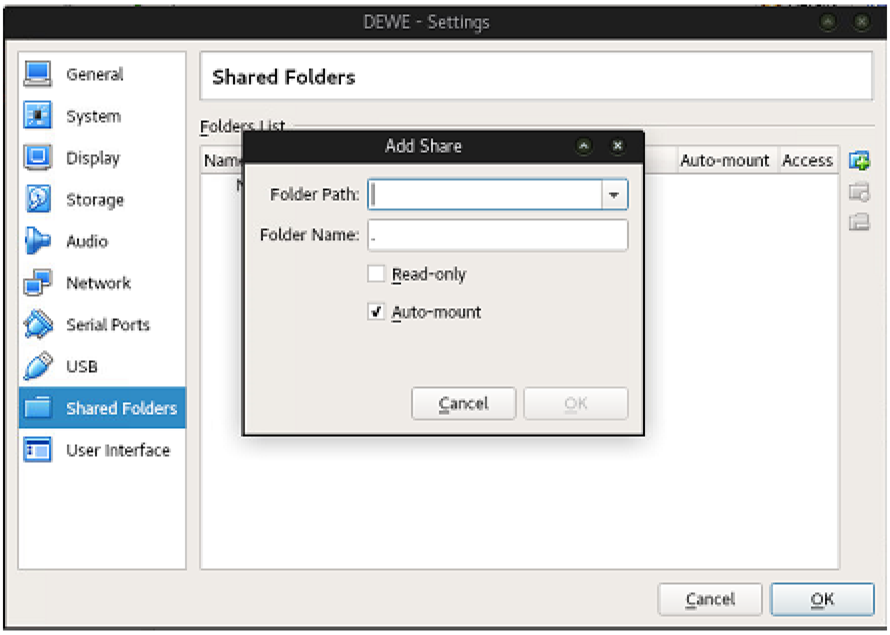
Network
Adapter 1: Intel PRO/1000 MT Desktop (NAT)

USB
USB Controller: OHCI

Click on the DEWE Virtual Machine *Settings* button. Here select the *Shared Folders* option, and click on add new shared folder icon (the icon that looks like a folder with a green plus symbol).



Select the path and name of the shared folder, and the *auto-mount* option.



Folder Path:

Folder Name: .

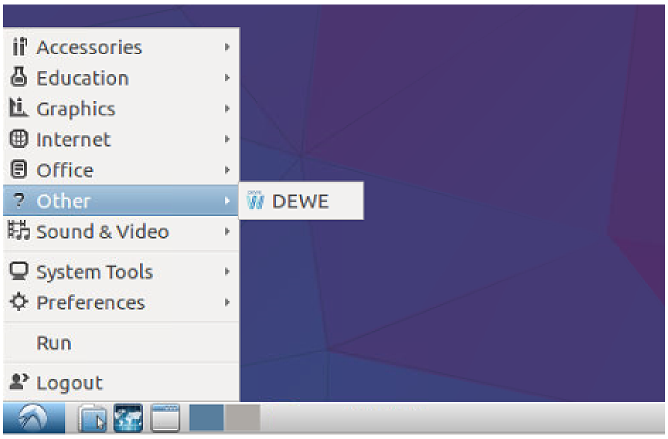
□ Read-only

☑ Auto-mount

Now, DEWE Virtual Machine is correctly installed and ready to use it.

2.2.4 First run

DEWE is now ready to start working. You can open it from the DEWE icon on the Start Menu:



Or typing on terminal:

```
$ dewe
```

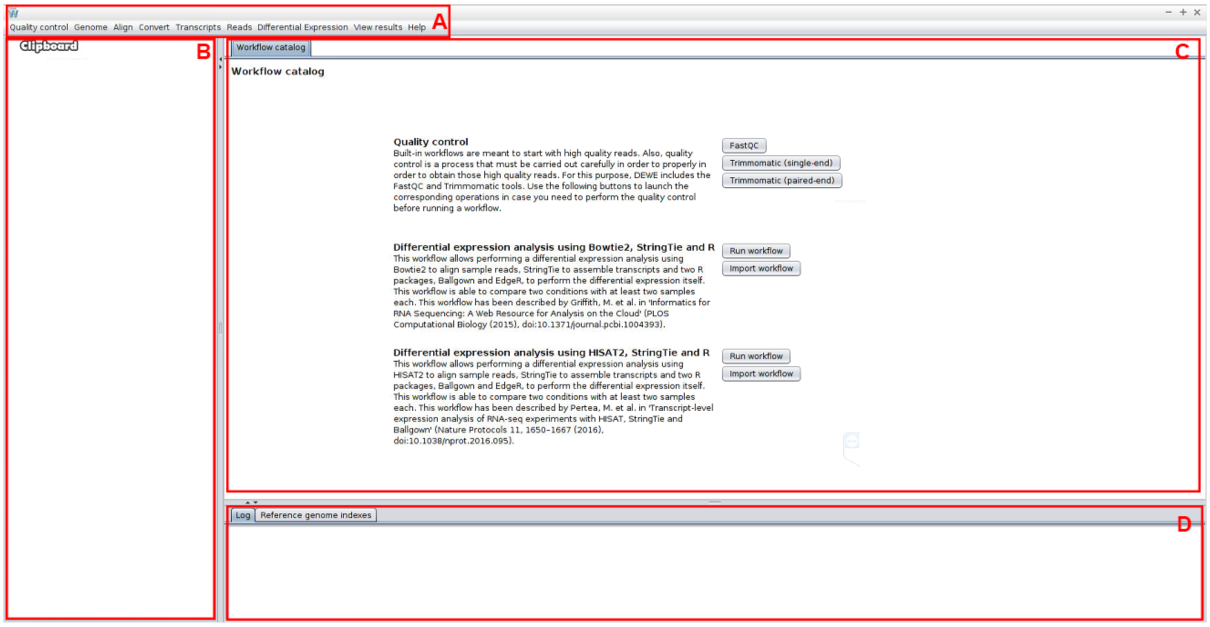
The virtual machine password is:

```
rnaseqpass
```

3. Graphical user interface

The Graphical User Interface (GUI) of DEWE has four main areas: (A) the *menu bar*, where the different DEWE functions are available (see section *5. Single Operations*), (B) the

Clipboard area, where final outputs are shown (see section *6. Outputs and visualization*), (C) the *view* area, where outputs can be inspected and the *Workflow catalog* (see section *4. DEWE workflows*) is shown when no outputs are being displayed, and (D) the *log* and *reference genomes index* area.



Workflow catalog

Quality control
Built-in workflows are meant to start with high quality reads. Also, quality control is a process that must be carried out carefully in order to properly in order to obtain those high quality reads. For this purpose, DEWE includes the FastQC and Trimmomatic tools. Use the following buttons to launch the corresponding operations in case you need to perform the quality control before running a workflow.

Differential expression analysis using Bowtie2, StringTie and R
This workflow allows performing a differential expression analysis using Bowtie2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Griffith, M. et al. in 'Informatics for RNA Sequencing: A Web Resource for Analysis on the Cloud' (PLOS Computational Biology (2015), doi:10.1371/journal.pcbi.1004393).

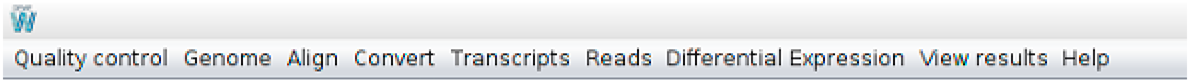
Differential expression analysis using HISAT2, StringTie and R
This workflow allows performing a differential expression analysis using HISAT2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Pertea, M. et al. in 'Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown' (Nature Protocols 11, 1650–1667 (2016), doi:10.1038/nprot.2016.095).

3.1 The *Menu bar* area

The *Menu bar* collects the entire single operations that DEWE can execute. This operations are:

- Quality control of samples.
- Build a reference index.
- Import a reference index.
- Align samples.
- Convert SAM to BAM files.
- Reconstruct transcripts.
- Merge transcripts.
- Calculate reads counts.
- Calculate differential expression.
- Visualise differential expression results.

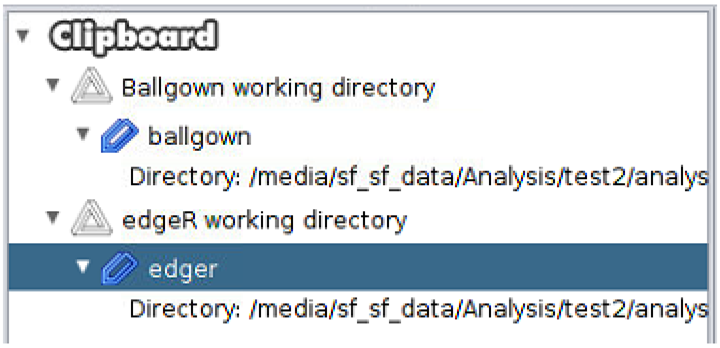
For a more detailed description of these operations, see section *5. Single Operations*.



3.2 The *Clipboard* area

The *Clipboard* area collects all the analysis that have been running since the application was opened, as well as those that have been imported through the *Menu* area.

For a detailed explanation about the DEWE outputs, see section *6. Outputs and visualisation.*



Ballgown working directory

ballgown

Directory: /media/sf_sf_data/Analysis/test2/analys

edgeR working directory

edger

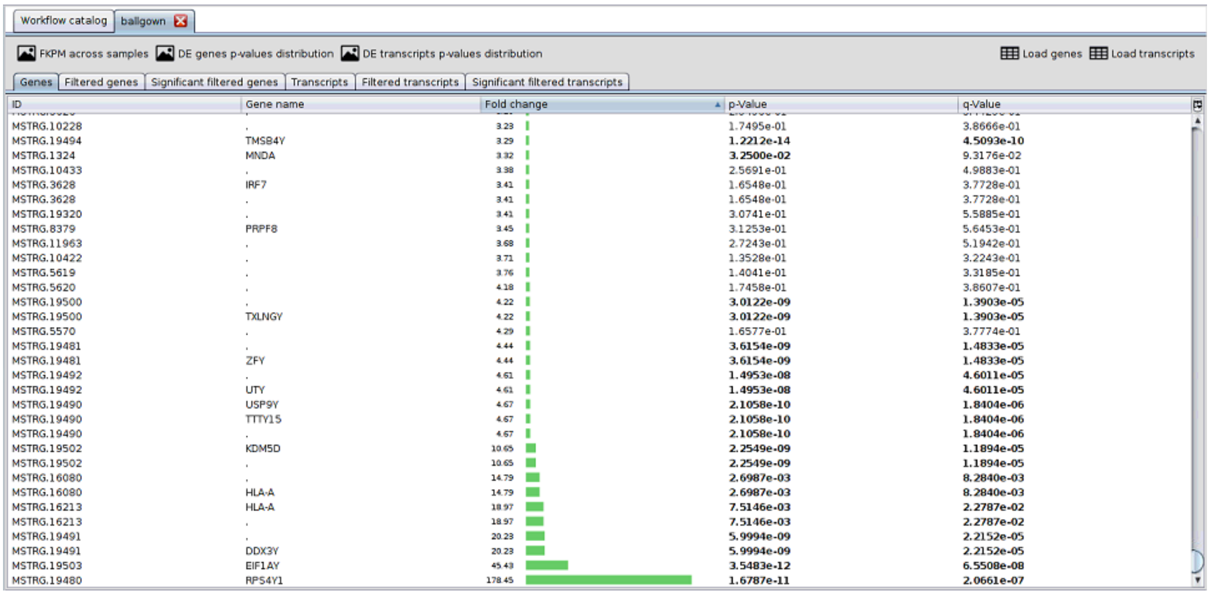
Directory: /media/sf_sf_data/Analysis/test2/analys

3.3 The View area

The *View* area contains on the one hand, the *Workflow catalog*, and on the other hand the inspector of the analysis opened on the *Clipboard* area.

The *Workflow catalog* collects the available RNA-seq workflows in DEWE. For detailed information about the available workflows, see section *4. DEWE workflows.*

The inspector of the opened analysis is a tool in charge of visualising the analysis executed in DEWE as well as of generating additional analysis. It is formed by all the tabs that are opening to the right of the *Workflow catalog* tab. For a detailed explanation about the DEWE analysis, see section *6. Outputs and visualisation.*

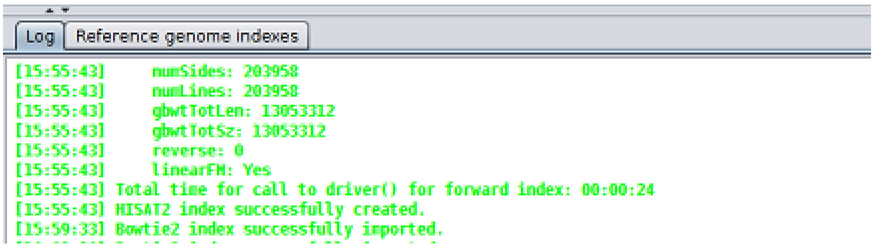

Workflow catalog
ballgown
FKPM across samples
DE genes p-values distribution
DE transcripts p-values distribution
Load genes
Load transcripts
Genes
Filtered genes
Significant filtered genes
Transcripts
Filtered transcripts
Significant filtered transcripts

ID	Gene name	Fold change	p-Value	q-Value
MSTRG.10228	.	3.23	1.7495e-01	3.8666e-01
MSTRG.19494	TMSB4Y	3.29	**1.2212e-14**	**4.5093e-10**
MSTRG.1324	MNDA	3.32	**3.2500e-02**	9.3176e-02
MSTRG.10433	.	3.38	2.5691e-01	4.9883e-01
MSTRG.3628	IRF7	3.41	1.6548e-01	3.7728e-01
MSTRG.3628	.	3.41	1.6548e-01	3.7728e-01
MSTRG.19320	.	3.41	3.0741e-01	5.5885e-01
MSTRG.8379	PRPF8	3.45	3.1253e-01	5.6453e-01
MSTRG.11963	.	3.68	2.7243e-01	5.1942e-01
MSTRG.10422	.	3.71	1.3528e-01	3.2243e-01
MSTRG.5619	.	3.76	1.4041e-01	3.3185e-01
MSTRG.5620	.	4.18	1.7458e-01	3.8607e-01
MSTRG.19500	.	4.22	**3.0122e-09**	**1.3903e-05**
MSTRG.19500	TXLNGY	4.22	**3.0122e-09**	**1.3903e-05**
MSTRG.5570	.	4.29	1.6577e-01	3.7774e-01
MSTRG.19481	.	4.44	**3.6154e-09**	**1.4833e-05**
MSTRG.19481	ZFY	4.44	**3.6154e-09**	**1.4833e-05**
MSTRG.19492	.	4.61	**1.4953e-08**	**4.6011e-05**
MSTRG.19492	UTY	4.61	**1.4953e-08**	**4.6011e-05**
MSTRG.19490	USP9Y	4.67	**2.1058e-10**	**1.8404e-06**
MSTRG.19490	TTTY15	4.67	**2.1058e-10**	**1.8404e-06**
MSTRG.19490	.	4.67	**2.1058e-10**	**1.8404e-06**
MSTRG.19502	KDM5D	10.65	**2.2549e-09**	**1.1894e-05**
MSTRG.19502	.	10.65	**2.2549e-09**	**1.1894e-05**
MSTRG.16080	.	14.79	**2.6987e-03**	**8.2840e-03**
MSTRG.16080	HLA-A	14.79	**2.6987e-03**	**8.2840e-03**
MSTRG.16213	HLA-A	18.97	**7.5146e-03**	**2.2787e-02**
MSTRG.16213	.	18.97	**7.5146e-03**	**2.2787e-02**
MSTRG.19491	.	20.23	**5.9994e-09**	**2.2152e-05**
MSTRG.19491	DDX3Y	20.23	**5.9994e-09**	**2.2152e-05**
MSTRG.19503	EIF1AY	45.43	**3.5483e-12**	**6.5508e-08**
MSTRG.19480	RPS4Y1	178.45	**1.6787e-11**	**2.0661e-07**

3.4 The *Log* area

The *Log* area collects the entire flow of events related to each of the operations that are executed in DEWE. Using this log, the user can follow the execution of the commands used by DEWE and also collect any error messages that may have been generated during the course of the execution of any of the operations. The execution log will be stored on

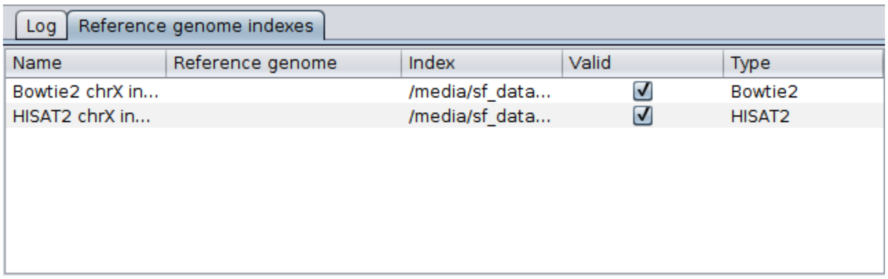
/mnt/shared/DEWE.log in the Docker versions, and on /opt/DEWE/DEWE.log in the Virtual Machine version.



```
[15:55:43]     numSides: 203958
[15:55:43]     numLines: 203958
[15:55:43]     gbwtTotLen: 13053312
[15:55:43]     gbwtTotSz: 13053312
[15:55:43]     reverse: 0
[15:55:43]     linearFM: Yes
[15:55:43] Total time for call to driver() for forward index: 00:00:24
[15:55:43] HISAT2 index successfully created.
[15:59:33] Bowtie2 index successfully imported.
```

3.5 The *Reference genome indexes* area

Some operations require a reference genome index to work (for example, in order to align RNA-Seq reads against a reference genome). The indexes can be built by DEWE from a reference genome, or they can be imported, that is, they have already been constructed before (inside or outside DEWE). The *Reference genome indexes* tab allows to manage the list of reference genome indexes currently import or built in DEWE. Please, refer to subsection *5.1 the Genome menu* in order to learn how to build or import new reference genome indexes.

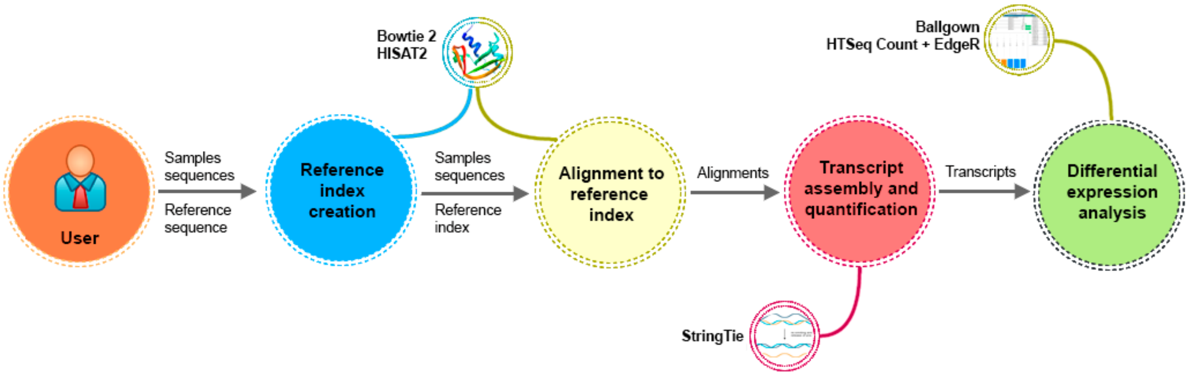


Name	Reference genome	Index	Valid	Type
Bowtie2 chrX in...		/media/sf_data...	☑	Bowtie2
HISAT2 chrX in...		/media/sf_data...	☑	HISAT2

4. DEWE workflows

Currently, *DEWE* provides two differential expression analysis workflows:

- Bowtie2, StringTie and R libraries (Ballgown and edgeR) [26].
- HISAT2, StringTie and R libraries (Ballgown and edgeR) [15].



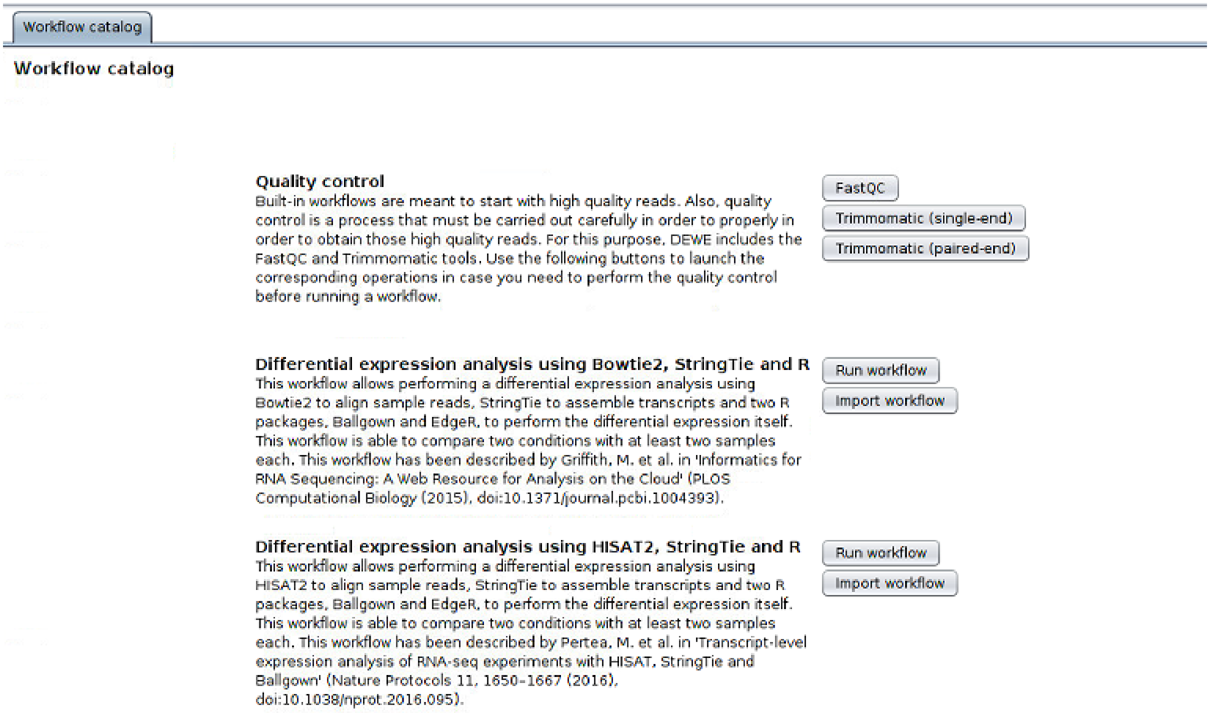
The two workflows implemented in DEWE align reads from RNA-seq experiments to reference genomes (RNA-seq data must come from eukaryotic organisms). One of the two DEWE pipelines is the current Tuxedo protocol which is, by far, the most used RNASeq analysis approach. Originally, the Tuxedo protocol, included the aligner software Tophat, and Cufflinks and Cuffdiff for differential gene expression estimation [16]. Improvements in the scaling and computational times led to a structural refactoring of the different Tuxedo modules to its actual configuration, which includes HISAT2, StringTie, Ballgown and edgeR [15]. In addition to facilitate analysis scaling, this protocol is claimed as the most accurate for the detection of differentially expressed genes. The other protocol contained in DEWE combines Bowtie2 as aligner, and Stringtie, Ballgown and edgeR for differential expression analysis [26]. This combination is less exigent from a computational point of view and can be more suitable for comparisons including smaller reference genomes. However, in terms of alignment, Bowtie2 was mainly designed to align samples without intron-sized gaps. The two-pass aligner HISAT2 better addresses this issue. Therefore, it is left to the user criteria whether apply HISAT2 or Bowtie2 to his/her analysis.

DEWE also includes two DE analysis libraries, Ballgown and edgeR, offering the end-user the possibility to compare gene/transcript expression results through two different normalisation procedures (Fragments Per Kilobase Million or FPKM vs Trimmed Mean of M-values or TMM). Although most of the DEWE outputs, mainly tables and figures, are originated from Ballgown, edgeR remains a tool widely used in the scientific community. For this reason, we consider interesting keeping this analysis and bringing the user the possibility to compare their results with many other pipelines that still use edgeR.

It is important to note that quality control operations must be performed manually by the user before the workflow execution, thus they are not included in the built-in workflows.

Additionaly, and for a detailed review of the basic conceps of RNA-Seq analysis, the reader is prompted to look up the Galaxy Project tutorial: http://galaxyproject.github.io/training-material/topics/transcriptomics/tutorials/rb-rnaseq/tutorial.html.

The available workflows are accessible from the *Workflow catalog*. This section contains tutorials for configure and execute the available workflows using real datasets.



Workflow catalog

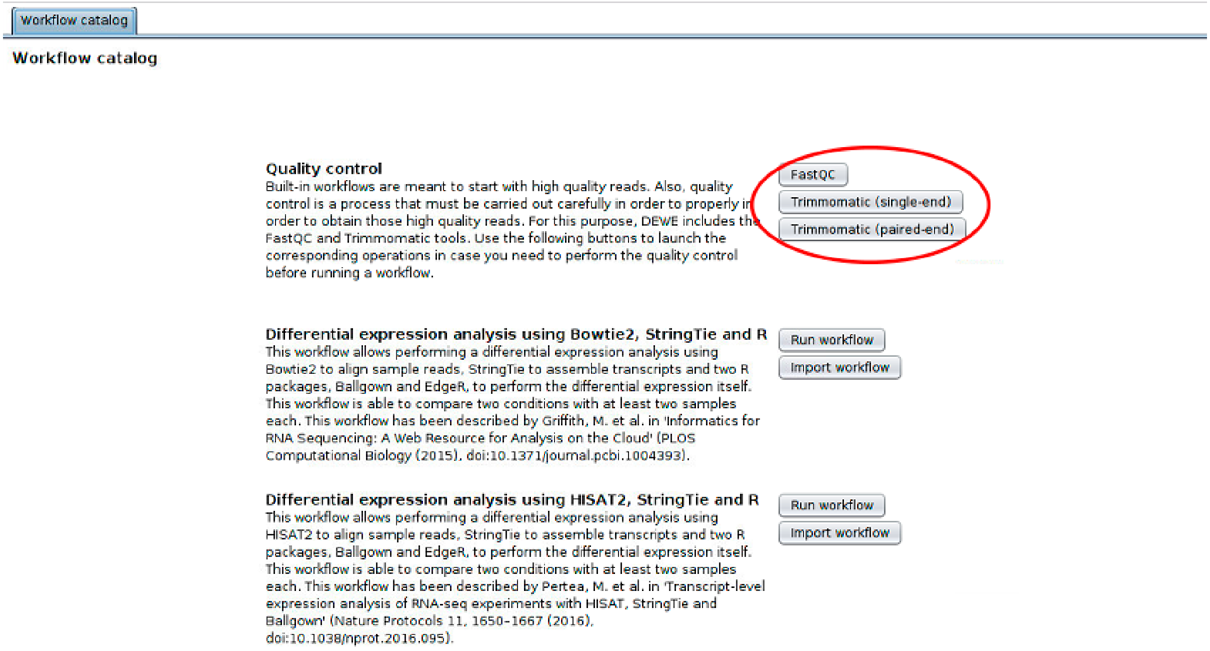
Quality control
Built-in workflows are meant to start with high quality reads. Also, quality control is a process that must be carried out carefully in order to properly in order to obtain those high quality reads. For this purpose, DEWE includes the FastQC and Trimmomatic tools. Use the following buttons to launch the corresponding operations in case you need to perform the quality control before running a workflow.

Differential expression analysis using Bowtie2, StringTie and R
This workflow allows performing a differential expression analysis using Bowtie2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Griffith, M. et al. in 'Informatics for RNA Sequencing: A Web Resource for Analysis on the Cloud' (PLOS Computational Biology (2015), doi:10.1371/journal.pcbi.1004393).

Differential expression analysis using HISAT2, StringTie and R
This workflow allows performing a differential expression analysis using HISAT2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Pertea, M. et al. in 'Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown' (Nature Protocols 11, 1650-1667 (2016), doi:10.1038/nprot.2016.095).

4.1. Quality control [Manual]

Before executing a workflow, users may need to perform quality control analysis and reads filtering in order to feed the workflows with proper data. For this task, DEWE offers the FastQC tool for the quality control of the samples, and Trimmomatic to perform reads filtering. It is important to note that these quality control operations must be performed manually by the user before the workflow execution, thus they are not included in the built-in workflows.



Workflow catalog

Quality control
Built-in workflows are meant to start with high quality reads. Also, quality control is a process that must be carried out carefully in order to properly in order to obtain those high quality reads. For this purpose, DEWE includes the FastQC and Trimmomatic tools. Use the following buttons to launch the corresponding operations in case you need to perform the quality control before running a workflow.

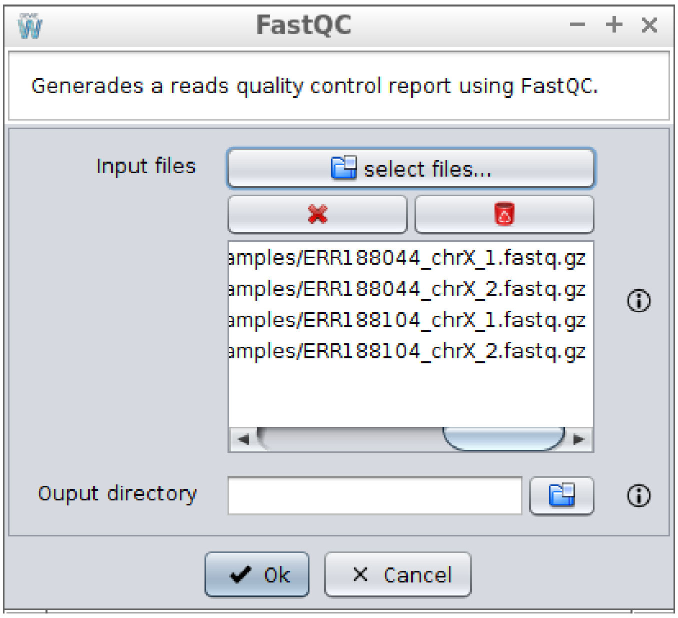
Differential expression analysis using Bowtie2, StringTie and R
This workflow allows performing a differential expression analysis using Bowtie2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Griffith, M. et al. in 'Informatics for RNA Sequencing: A Web Resource for Analysis on the Cloud' (PLOS Computational Biology (2015). doi:10.1371/journal.pcbi.1004393).

Differential expression analysis using HISAT2, StringTie and R
This workflow allows performing a differential expression analysis using HISAT2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Pertea, M. et al. in 'Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown' (Nature Protocols 11, 1650-1667 (2016). doi:10.1038/nprot.2016.095).

4.1.1 FastQC

DEWE allows the generation of a FastQC quality control report for multiple reads files. Clicking on the *FastQC* button in the *Workflow catalog*, a new window will be displayed and the following data will be requested:

- *Input files*: the reference annotation file (.gtf).
- *Output directory*: optionally, the directory where the reports must be generated. If not provided, the output report for each reads file is created in the same directory as the reads file being processed.



Generades a reads quality control report using FastQC.

Input files
amples/ERR188044_chrX_1.fastq.gz
amples/ERR188044_chrX_2.fastq.gz
amples/ERR188104_chrX_1.fastq.gz
amples/ERR188104_chrX_2.fastq.gz

Ouput directory

Once the *Ok* button is pressed, StringTie analysis starts and a message will be displayed until the end of the process, when an information message is shown.

4.1.2 Trimmomatic

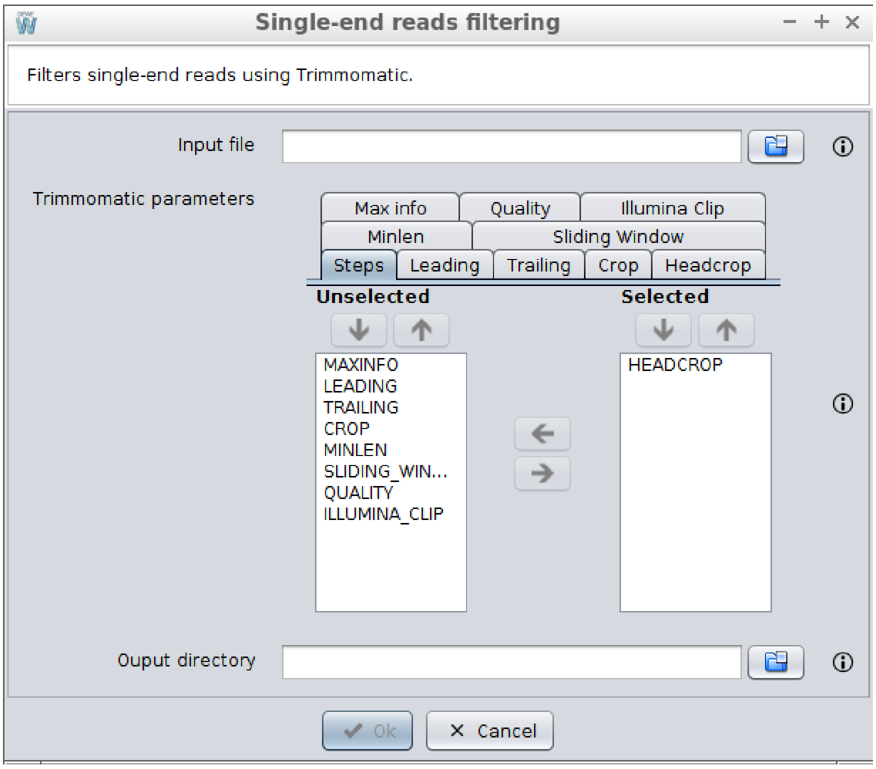
DEWE provides operations for performing reads filtering using Trimmomatic.

4.1.2.1 Single-end reads filtering

This operation allows filtering single-end raw reads using Trimmomatic. Clicking on the *Trimmomatic (single-end)* button in the *Workflow catalog*, a new window will be displayed and the following data will be requested:

- *Input file*: the input reads file.
- *Trimmomatic parameters*: the steps for trimmomatic and its configuration. The *Steps* tab allows selecting which steps must be applied and define the order in which they should be applied. Then, the other tabs allows configuring each step. The following steps are available:
  - Leading: removes low quality bases from the beginning. As long as a base has a value below this threshold the base is removed and the next base will be investigated.
  - Trailing: removes low quality bases from the end. As long as a base has a value below this threshold the base is removed and the next base (which as trimmomatic is starting from the 3' prime end would be base preceding the just removed base) will be investigated. This approach can be used removing the special illumina 'low quality segment' regions (which are marked with quality score of 2), but we recommend Sliding Window or MaxInfo instead.
  - Crop: removes bases regardless of quality from the end of the read, so that the read has maximally the specified length after this step has been performed. Steps performed after CROP might of course further shorten the read.
  - Headcrop: removes the specified number of bases, regardless of quality, from the beginning of the read.
  - Minlen: removes reads that fall below the specified minimal length. If required, it should normally be after all other processing steps. Reads removed by this step will be counted and included in the „dropped reads" count presented in the trimmomatic summary.
  - Sliding window: performs a sliding window trimming, cutting once the average quality within the window falls below a threshold. By considering multiple bases, a single poor quality base will not cause the removal of high quality data later in the read.
  - Max info: performs an adaptive quality trim, balancing the benefits of retaining longer reads against the costs of retaining bases with errors.
  - Quality: reencodes the quality part of the FASTQ file to the selected base.
  - Illumina clip: finds and removes Illumina adapters.

- *Output directory*: optionally, the directory where the filtered file must be created. If not provided, the output file is created in the same directory as the reads file being filtered.



Filters single-end reads using Trimmomatic.

Unselected
MAXINFO
LEADING
TRAILING
CROP
MINLEN
SLIDING_WIN...
QUALITY
ILLUMINA_CLIP

Selected
HEADCROP

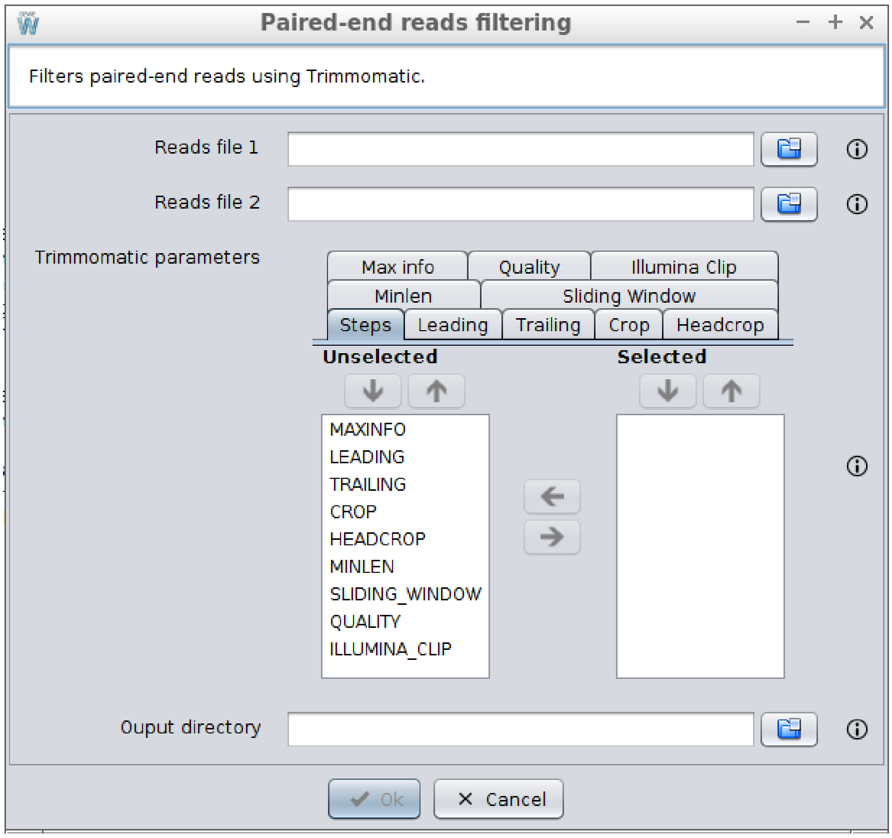
4.1.2.2 Paired-end reads filtering

This operation allows filtering paired-end raw reads using Trimmomatic. Clicking on the *Trimmomatic (paired-end)* button in the *Workflow catalog*, a new window will be displayed and the following data will be requested:

- *Reads file 1*: the first reads file.
- *Reads file 2*: the second reads file
- *Trimmomatic parameters*: the steps for trimmomatic and its configuration. The *Steps* tab allows selecting which steps must be applied and define the order in which they should be applied. Then, the other tabs allows configuring each step. The following steps are available:
    - Leading: removes low quality bases from the beginning. As long as a base has a value below this threshold the base is removed and the next base will be investigated.
    - Trailing: removes low quality bases from the end. As long as a base has a value below this threshold the base is removed and the next base (which as trimmomatic is starting from the 3' prime end would be base preceding the just removed base) will be investigated. This approach can be used removing the special illumina 'low quality segment' regions (which are marked with quality score of 2), but we recommend Sliding Window or MaxInfo instead.
    - Crop: removes bases regardless of quality from the end of the read, so that the read has maximally the specified length after this step has been

performed. Steps performed after CROP might of course further shorten the read.

  - Headcrop: removes the specified number of bases, regardless of quality, from the beginning of the read.
  - Minlen: removes reads that fall below the specified minimal length. If required, it should normally be after all other processing steps. Reads removed by this step will be counted and included in the „dropped reads" count presented in the trimmomatic summary.
  - Sliding window: performs a sliding window trimming, cutting once the average quality within the window falls below a threshold. By considering multiple bases, a single poor quality base will not cause the removal of high quality data later in the read.
  - Max info: performs an adaptive quality trim, balancing the benefits of retaining longer reads against the costs of retaining bases with errors.
  - Quality: reencodes the quality part of the FASTQ file to the selected base.
  - Illumina clip: finds and removes Illumina adapters.
- *Output directory*: optionally, the directory where the filtered files must be created. If not provided, the output files are created in the same directory as the reads file being filtered.



Paired-end reads filtering

Filters paired-end reads using Trimmomatic.

Reads file 1

Reads file 2

Trimmomatic parameters

Unselected

MAXINFO
LEADING
TRAILING
CROP
HEADCROP
MINLEN
SLIDING_WINDOW
QUALITY
ILLUMINA_CLIP

Selected

Ouput directory

4.2 Bowtie2, StringTie and R libraries (Ballgown and edgeR)

This workflow was introduced by Griffith, M. et al. [26]. As the title suggests, this workflow makes use of the tools Bowtie2 [5] to align sample reads, StringTie [7] to assemble transcripts and two R libraries, Ballgown [10] and edgeR [11], to perform the differential expression itself:

- **Bowtie2** aligns RNA-Seq reads to a genome. This aligner is less exigent than HISAT2 from a computational point of view and can be more suitable for comparisons including smaller reference genomes. However, Bowtie2 was mainly designed to align samples without intron-sized gaps.
- **StringTie** assembles the alignments into full and partial transcripts, creating multiple isoforms as necessary and estimating the expression levels of all genes and transcripts. StringTie normalises the sequence depth and gene length by reporting the quantification results in FPKM (Fragments Per Kilobase Million) and in TPM (Transcripts Per kilobase Million).
- **Ballgown** takes the transcripts and expression levels from StringTie normalised in FPKM and applies rigorous statistical methods to determine which transcripts are differentially expressed between the conditions. Besides, **edgeR** uses raw count produced by HtSeq and then normalises this raw counts in TMM (Trimmed Mean of M-values). A detailed explanation between how the different normalization metrics work, can be found in this excellent online resource: https://www.rna-seqblog.com/rpkm-fpkm-and-tpm-clearly-explained/

The workflow is illustrated with the sample HCC1395 that contains mammary gland samples from a female *H. sapiens*. The example data used in this workflow comprise human RNA-Seq samples, but to make this execution faster and simpler for testing, a subset of the reads mapping to human chromosome 22 has been extracted. The conditions to be compared will be normal (this being the control) and tumor (this being the case) samples, and there are three samples for each condition (Table 2). The RNA-Seq samples will be aligned against the chromosome 22 of the *H. sapiens* and also annotated against the chromosome 22.

Table 2: Condition for each sample used in the workflow.

Sample id	**Condition**
hcc1395_normal_rep1	Normal
hcc1395_normal_rep2	Normal
hcc1395_normal_rep3	Normal
hcc1395_tumor_rep1	Tumor
hcc1395_tumor_rep2	Tumor

hcc1395_tumor_rep3	Tumor

Step 1: download the dataset

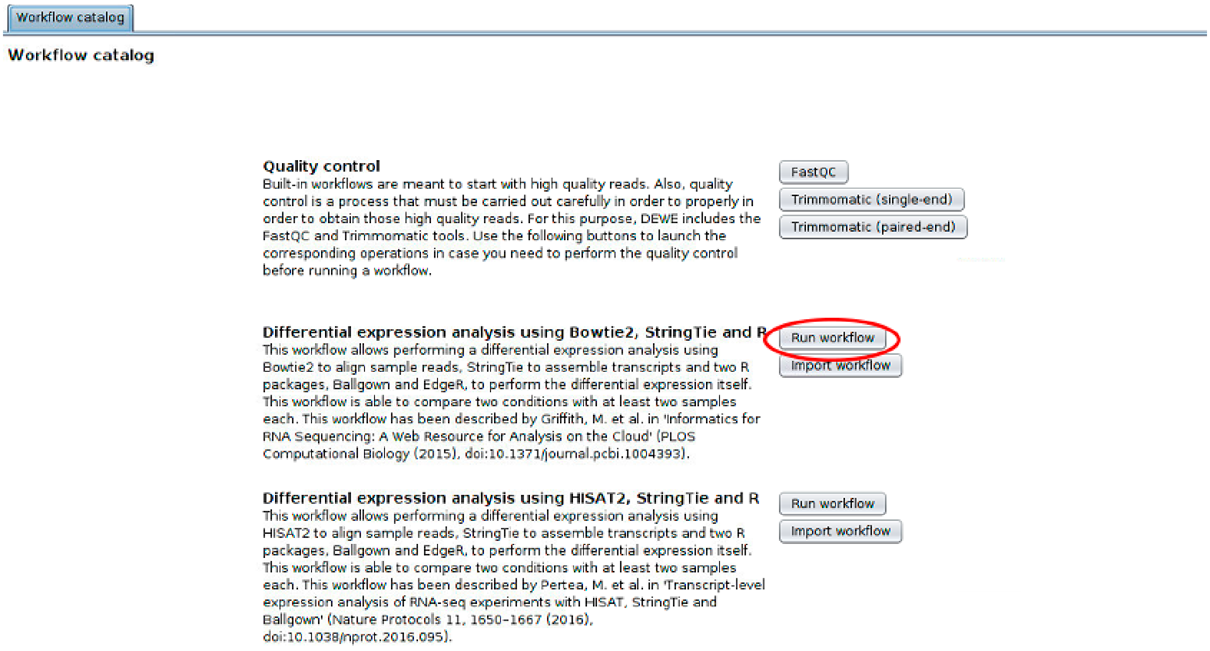
The example dataset is available at the following URL: http://static.sing-group.org/software/DEWE/data/tutorial_data_HCC1395.zip. This dataset must be downloaded and uncompressed in the application shared folder.

The dataset contains the following files and directories:

- *genes*: a directory containing the reference annotation file called *Homo_sapiens.GRCh38.86.chromosome22.gtf*.
- *genome*: a directory containing the reference genome in fasta format.
- *indexes*: a directory containing the Bowtie2 indexes of the chromosome 22 reference genome.
- *samples*: a directory containing two folders:
  - *normal*: a directory containing the paired-end reads corresponding to the 3 normal samples in the dataset.
  - *tumor*: a directory containing the paired-end reads corresponding to the 3 tumor samples in the dataset.

Step 2: configure the workflow

The next step consists in configuring the workflow. To do so, go to the *Workflow catalog* and click the *Run workflow* button next to the *Bowtie2, StringTie and R* workflow description.



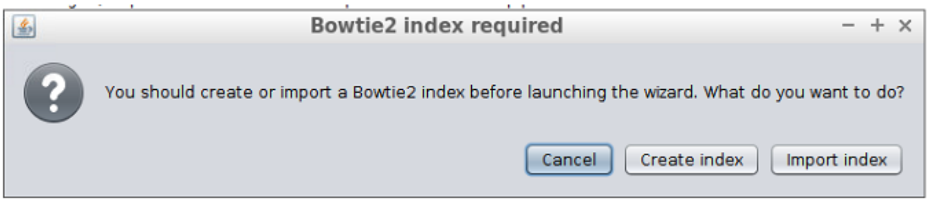
Workflow catalog

Quality control
Built-in workflows are meant to start with high quality reads. Also, quality control is a process that must be carried out carefully in order to properly in order to obtain those high quality reads. For this purpose, DEWE includes the FastQC and Trimmomatic tools. Use the following buttons to launch the corresponding operations in case you need to perform the quality control before running a workflow.

Differential expression analysis using Bowtie2, StringTie and R
This workflow allows performing a differential expression analysis using Bowtie2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Griffith, M. et al. in 'Informatics for RNA Sequencing: A Web Resource for Analysis on the Cloud' (PLOS Computational Biology (2015), doi:10.1371/journal.pcbi.1004393).

Differential expression analysis using HISAT2, StringTie and R
This workflow allows performing a differential expression analysis using HISAT2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Pertea, M. et al. in 'Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown' (Nature Protocols 11, 1650-1667 (2016), doi:10.1038/nprot.2016.095).

When the workflow is executed for the first time or no Bowtie2 reference genome indexes are available, the tool requires the importation or creation of a reference genome index using Bowtie2.


Bowtie2 index required

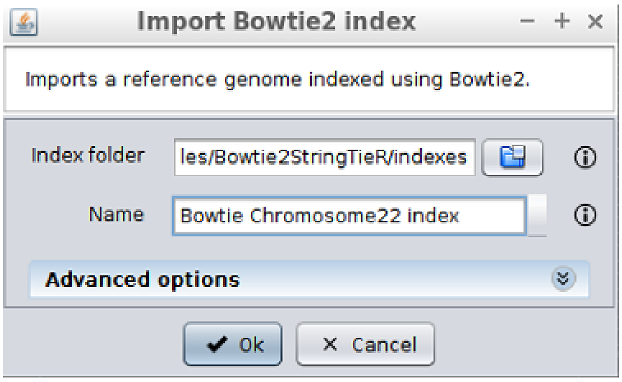
You should create or import a Bowtie2 index before launching the wizard. What do you want to do?

To build a new index, click the *Create index* button and proceed as explained in section 5.1.1.1. If the reference genome index already exists, as provided in the case study, click the *Import index* button.

Step 3: import the reference genome index

After clicking the *Import index* button, the following dialog will appear, allowing to select the downloaded reference genome index.The following data will be requested:

- *Index folder*: the directory that contains the Bowtie2 genome index. Select the *indexes* folder in the case study data. When selecting the folder, the files it contains will not be displayed.
- *Name*: the name for the genome index in order to identify it later.

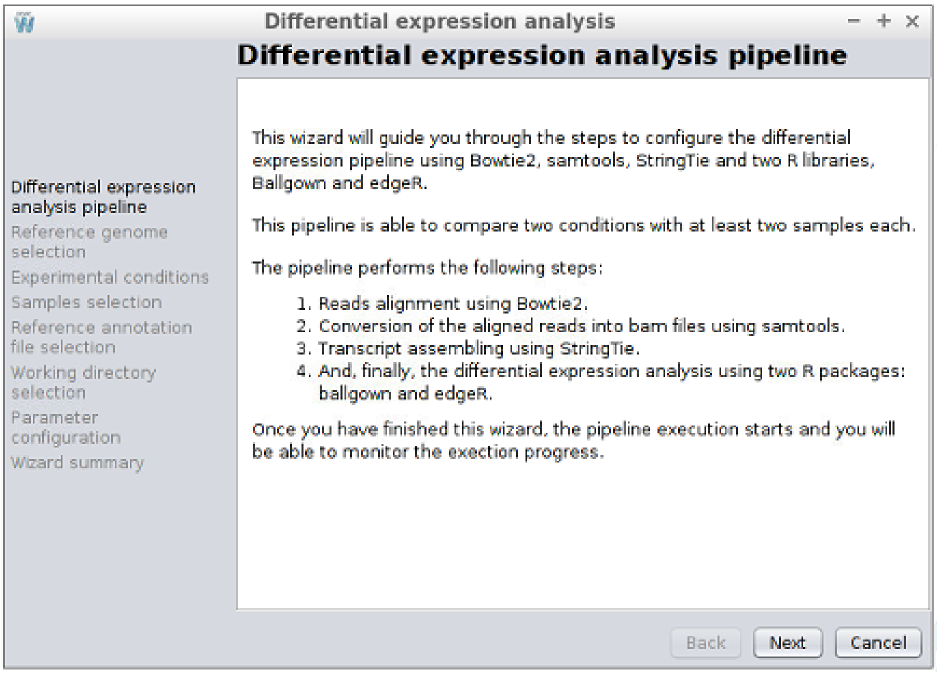

Import Bowtie2 index

Imports a reference genome indexed using Bowtie2.

Index folder: les/Bowtie2StringTieR/indexes

Name: Bowtie Chromosome22 index

After setting the required data, click the *Ok* button in order to import the reference genome index. Once the index is successfully imported, it is automatically added to the reference genome indexes database and the workflow configuration assistant opens. The advanced options for import index operation are explained in section 5.2.2.1.



Differential expression analysis
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Back
Next
Cancel

Differential expression analysis pipeline

This wizard will guide you through the steps to configure the differential expression pipeline using Bowtie2, samtools, StringTie and two R libraries, Ballgown and edgeR.

This pipeline is able to compare two conditions with at least two samples each.

The pipeline performs the following steps:

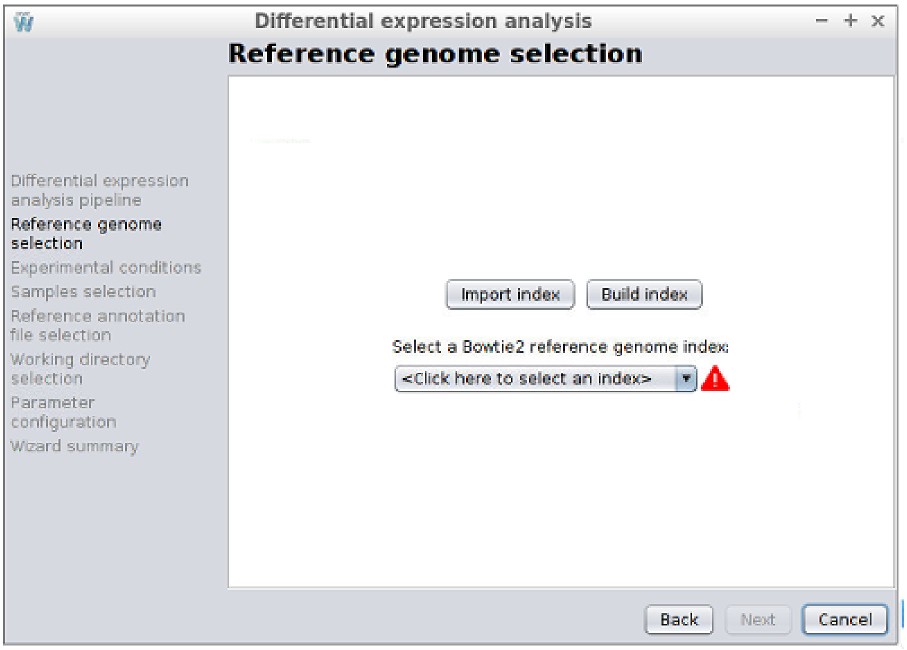
1. Reads alignment using Bowtie2.
2. Conversion of the aligned reads into bam files using samtools.
3. Transcript assembling using StringTie.
4. And, finally, the differential expression analysis using two R packages: ballgown and edgeR.

Once you have finished this wizard, the pipeline execution starts and you will be able to monitor the exection progress.

Click the *Next* button to advance to the next step.

Step 4: reference genome selection

In this step, the reference genome to be used to perform the alignment must be chosen. As shown in the following image, the configuration assistant shows the available Bowtie2 indexes. Note that in this step you can also use the *Import index* and *Build index* buttons to import or create a new reference genome index.

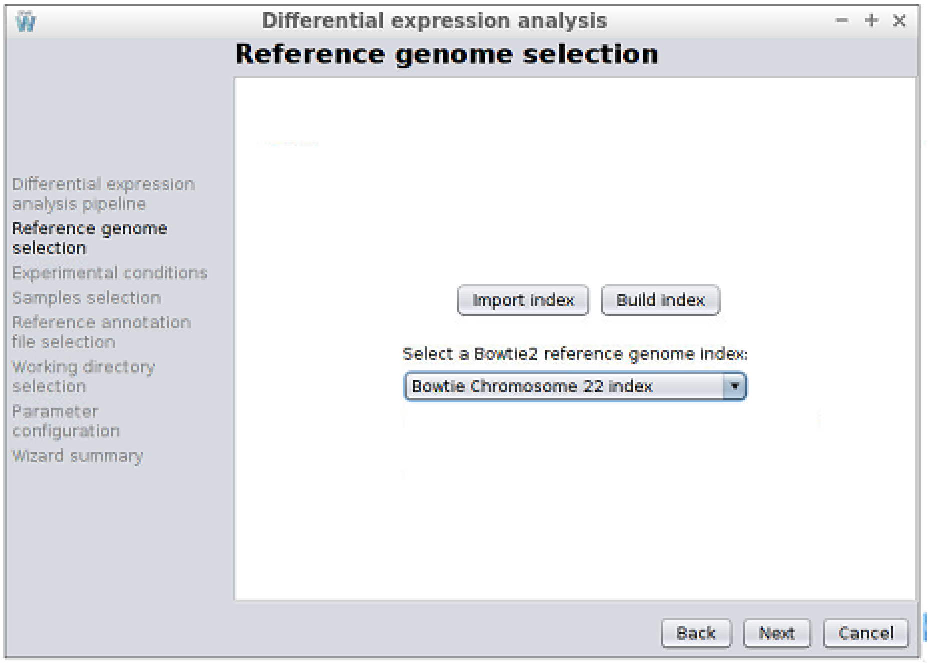


Differential expression analysis
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Import index
Build index
<Click here to select an index>
Back
Next
Cancel

Reference genome selection

Select a Bowtie2 reference genome index:

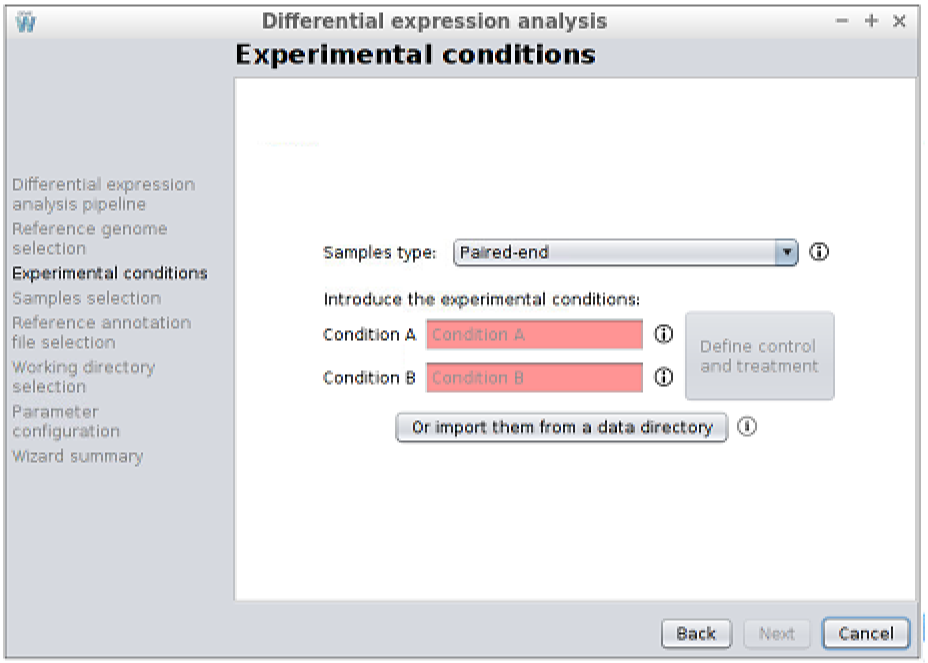
Select the index imported as shown in the image below and click the *Next* button to advance to the next step.


Differential expression analysis
Reference genome selection
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Import index
Build index
Back
Next
Cancel

Select a Bowtie2 reference genome index: Bowtie Chromosome 22 index

Step 5: introducing the experimental conditions

In this step, the names of the two experimental conditions of the experiment must be introduced.


Differential expression analysis
Experimental conditions
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Define control and treatment
Or import them from a data directory
Back
Next
Cancel

Samples type: Paired-end

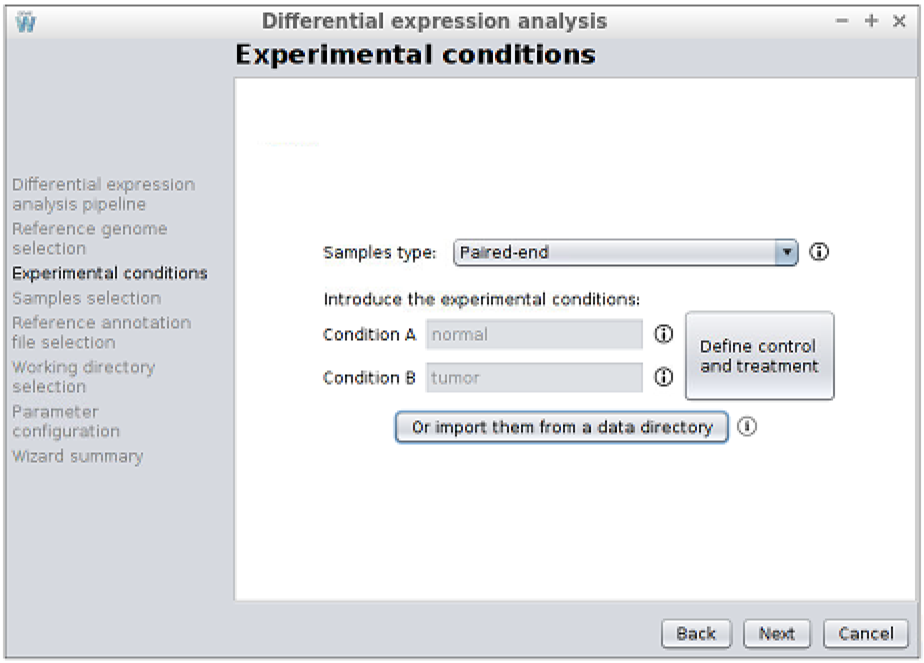
Introduce the experimental conditions:

Condition A: Condition A

Condition B: Condition B

DEWE allows to enter conditions and samples in a simple way by importing them from a directory (by default, paired-end samples type is selected). To do this, click on the "Or import data from a directory" button and select the *samples* the directory where the samples are located so that DEWE can automatically import all the data. Note that this directory should be organized as follows:

- It must contain two folders and the name of each folder will correspond to a condition. In this case, the *samples* folder contains the *normal* and *tumor* folders.
- Each of the two folders must contain the pairwise files of the samples and these files must be in .fq, .fastq or .fastq.gz format. In addition, as samples are paired-end, the first reads file must end in _1 and the second in _2.


Differential expression analysis
Experimental conditions
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Define control and treatment
Or import them from a data directory
Back
Next
Cancel

Samples type: Paired-end

Introduce the experimental conditions:

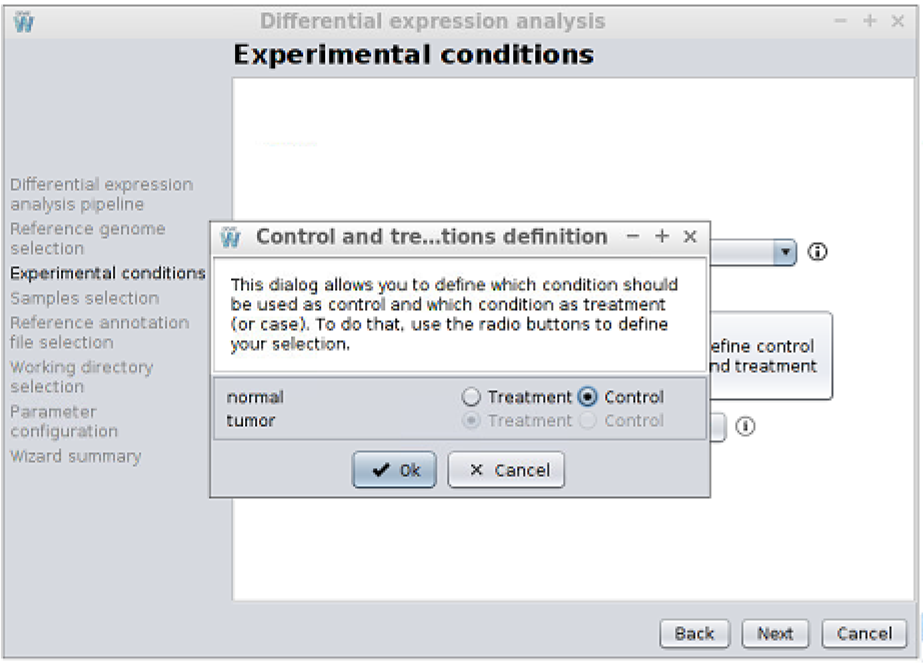
Condition A: normal

Condition B: tumor

To manually enter the data, the *normal* and *tumoral* conditions must be entered and then click on the *Next* button to advance to the next step. The order here is not important, but keep in mind that the control condition will be the first alphabetically.

Advanced: Define control and treatment condition

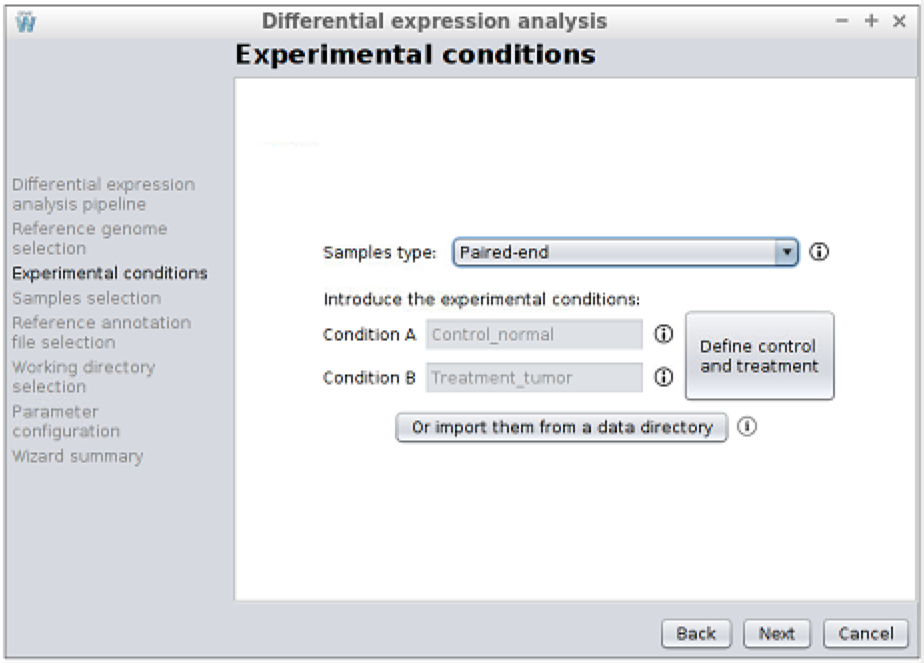
DEWE determines the control condition as the first alphabetical ordered regardless of the order in which they were entered. To change this, there is a button at the right of the conditions text boxes, "Define control and treatment ", which opens a new window where conditions can be defined as Control and Treatment.


Differential expression analysis
Experimental conditions
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Control and tre...tions definition
efine control
nd treatment
✔ Ok
✕ Cancel
Back
Next
Cancel

This dialog allows you to define which condition should be used as control and which condition as treatment (or case). To do that, use the radio buttons to define your selection.

normal ○ Treatment ● Control
tumor ● Treatment ○ Control

Once defined and pressed the "Ok" button, DEWE updates the text boxes, with the text “Control_” in front of the control condition name and “Treatment_” in front of the treatment condition name.


Differential expression analysis
Experimental conditions
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Define control and treatment
Or import them from a data directory
Back
Next
Cancel

Samples type: Paired-end

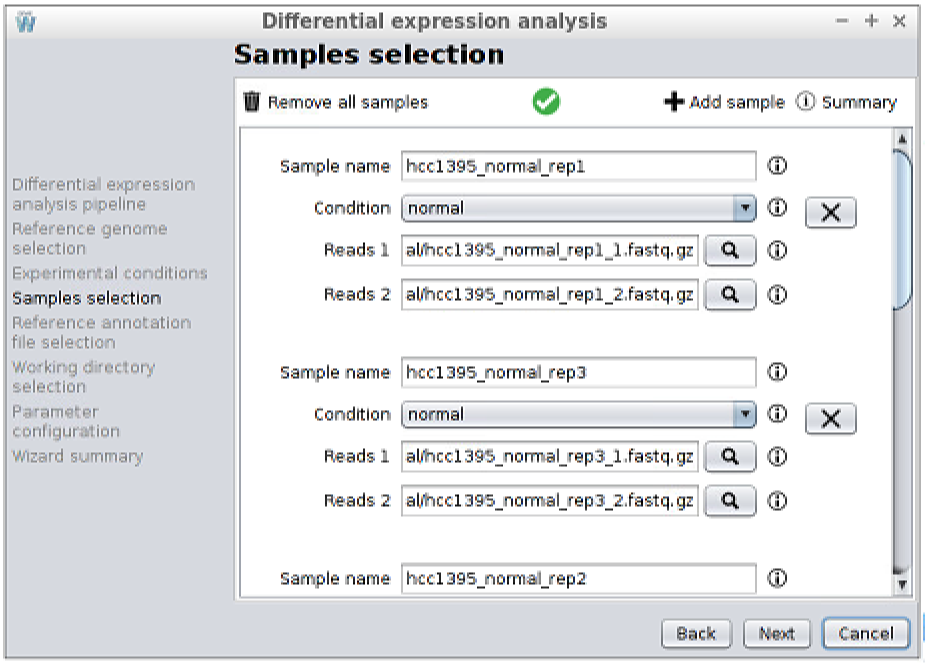
Introduce the experimental conditions:

Condition A Control_normal

Condition B Treatment_tumor

Step 6: samples verification *[Optional: samples selection]*

If the samples directory was selected previously, this step can be used to verify that all samples have been selected correctly.



Differential expression analysis

Samples selection

Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Remove all samples
Add sample
Summary
Back
Next
Cancel

Sample name: hcc1395_normal_rep1
Condition: normal
Reads 1: al/hcc1395_normal_rep1_1.fastq.gz
Reads 2: al/hcc1395_normal_rep1_2.fastq.gz

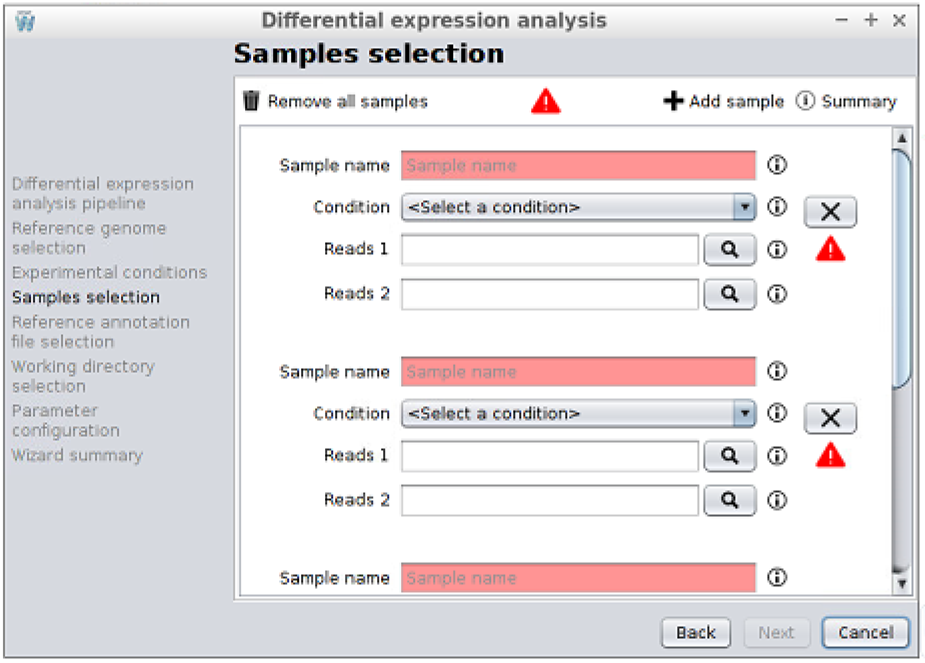
Sample name: hcc1395_normal_rep3
Condition: normal
Reads 1: al/hcc1395_normal_rep3_1.fastq.gz
Reads 2: al/hcc1395_normal_rep3_2.fastq.gz

Sample name: hcc1395_normal_rep2

Check if all samples are correctly imported and click the *Next* button to advance to the next step.

Optional: Samples selection

If a samples directory has not been selected in the previous step, the samples must be entered manually (DEWE allows sample files in .fq, .fastq or .fastq.gz format):



Differential expression analysis

Samples selection

Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Remove all samples
Add sample
Summary
Back
Next
Cancel

Sample name: Sample name
Condition: <Select a condition>
Reads 1:
Reads 2:

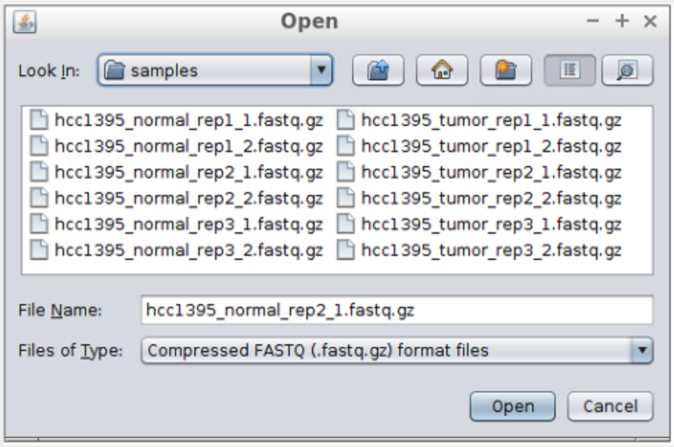
Sample name: Sample name
Condition: <Select a condition>
Reads 1:
Reads 2:

Sample name: Sample name

In order to introduce the samples for the tutorial dataset, follow the next steps:

1. Select the first reads file of each sample (those ending with *_1.fastq.gz* in the *samples* directory) by clicking the 🔍 button.

- Reads are provided in compressed FASTQ format so if the file browser is empty when browsing into the *samples* directory, then the appropriate file filter must be select as the image below shows.



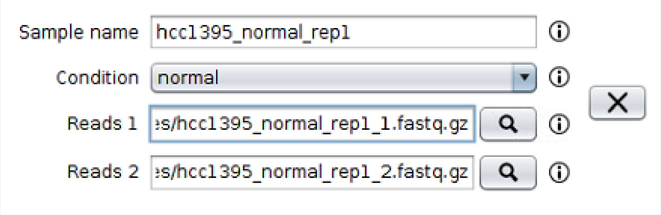
samples

hcc1395_normal_rep1_1.fastq.gz
hcc1395_normal_rep1_2.fastq.gz
hcc1395_normal_rep2_1.fastq.gz
hcc1395_normal_rep2_2.fastq.gz
hcc1395_normal_rep3_1.fastq.gz
hcc1395_normal_rep3_2.fastq.gz
hcc1395_tumor_rep1_1.fastq.gz
hcc1395_tumor_rep1_2.fastq.gz
hcc1395_tumor_rep2_1.fastq.gz
hcc1395_tumor_rep2_2.fastq.gz
hcc1395_tumor_rep3_1.fastq.gz
hcc1395_tumor_rep3_2.fastq.gz

File Name: hcc1395_normal_rep2_1.fastq.gz

Files of Type: Compressed FASTQ (.fastq.gz) format files

- As can be seen in the image below, after selecting the first reads file the second reads file and the sample name are automatically filled by the tool.



Sample name: hcc1395_normal_rep1

Condition: normal

Reads 1: :s/hcc1395_normal_rep1_1.fastq.gz

Reads 2: :s/hcc1395_normal_rep1_2.fastq.gz

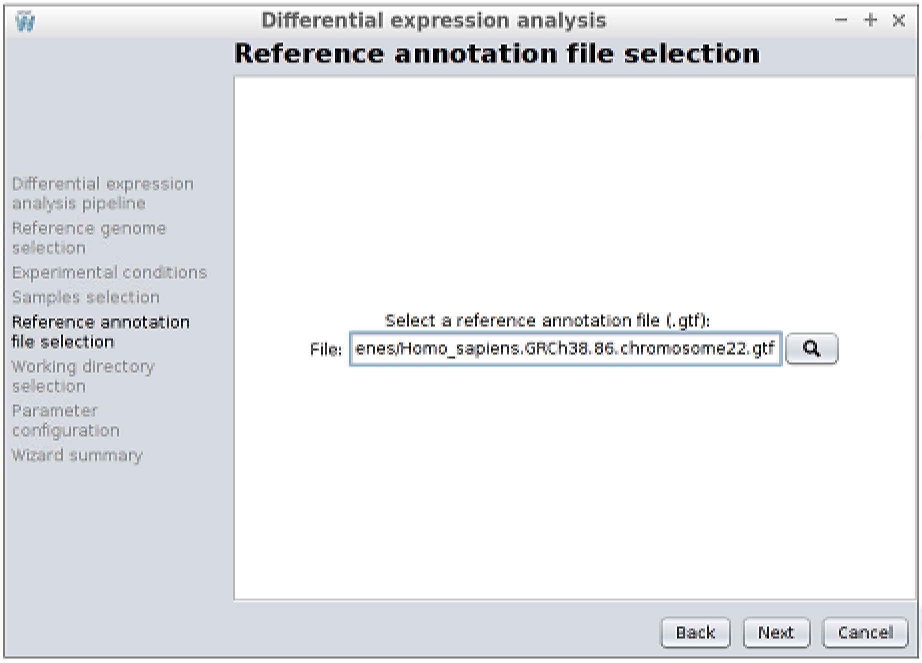
- By default, the list contains four samples. Since in this case you must introduce up to twelve samples, the **+ Add sample** button must be clicked to introduce more samples.

2. Assign each of the 6 samples to their corresponding conditions:
   - hcc1395_normal_rep1: normal.
   - hcc1395_normal_rep2: normal.
   - hcc1395_normal_rep3: normal.
   - hcc1395_tumor_rep1: tumor.
   - hcc1395_tumor_rep2: tumor.
   - hcc1395_tumor_rep3: tumor.

After introducing all the samples proceed to the next step. For single-end samples alignment the steps are the same, but only one single-end file is selected for each sample.

Step 7: reference annotation file selection

Select the reference annotation file (GTF format) for the experiment. In this case, the *Homo_sapiens.GRCh38.86.chromosome22.gtf* file located in the *genes* directory must be selected. After selecting it, the *Next* button must be clicked to advance to the next step. A ready-to-use reference sequence list and their annotations can be downloaded form the Illumina iGenome site (https://support.illumina.com/sequencing/sequencing_software/igenome.html), taking into account that only eukaryotic organisms can be analysed through DEWE.

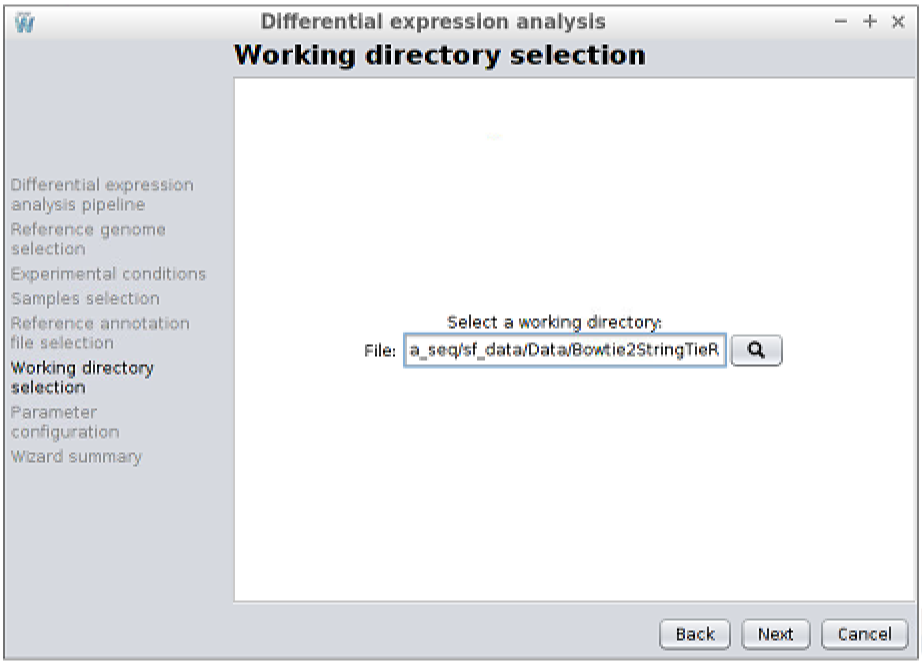

Differential expression analysis
Reference annotation file selection
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Back
Next
Cancel

Select a reference annotation file (.gtf):

File: enes/Homo_sapiens.GRCh38.86.chromosome22.gtf

Step 8: working directory selection

Choose the working directory, where the analysis results will be stored, and advance to the final step.


Differential expression analysis
Working directory selection
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Back
Next
Cancel

Select a working directory:

File: a_seq/sf_data/Data/Bowtie2StringTieR

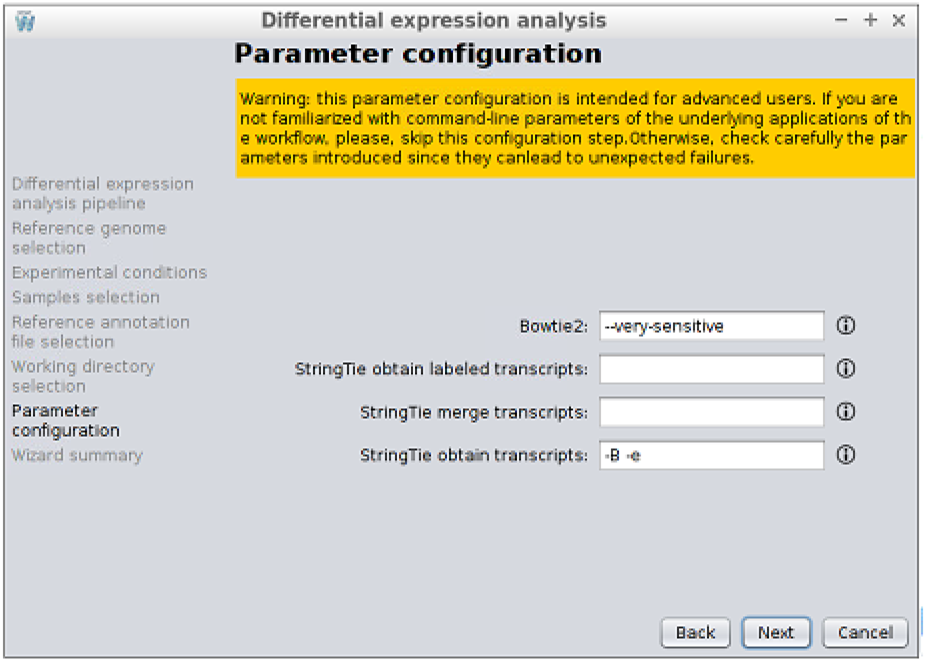
Step 9: parameter configuration

WARNING: this parameter configuration is intended for advanced users. If you are not familiarized with command-line parameters of the underlying applications of the workflow, please, skip this configuration step. Otherwise, check carefully the parameters introduced since they can lead to unexpected failures.

DEWE allows the user to manually introduce additional parameters to the alignment and transcript reconstruction steps, like a command line execution.

- *Bowtie2*: custom command-line parameters for the execution of the Bowtie2 alignment. For more information on Bowtie2 options, please, check the reference manual (http://bowtie-bio.sourceforge.net/bowtie2/manual.shtml#options) and sections 5.3.1.1 and 5.3.2.1.
- *StringTie obtain labeled transcripts*: custom command-line parameters for the execution of StringTie reconstruct labeled transcripts command. For more information on StrinTie options, please, check the reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual) and section 5.5.2 for more details.
- *StringTie merge transcripts*: custom command-line parameters for the execution of the StringTie merge command. For more information on StrinTie options, please, check the reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual) and section 5.5.3 for more details.
- *StringTie obtain transcripts*: custom command-line parameters for the execution of StringTie reconstruct transcripts command. For more information on StrinTie options, please, check the reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual) and section 5.5.1 for more details.

It is important that the parameters are entered correctly or the execution of their respective step will fail.



Differential expression analysis
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Back
Next
Cancel

Parameter configuration

Warning: this parameter configuration is intended for advanced users. If you are not familiarized with command-line parameters of the underlying applications of the workflow, please, skip this configuration step.Otherwise, check carefully the parameters introduced since they canlead to unexpected failures.

Bowtie2: --very-sensitive

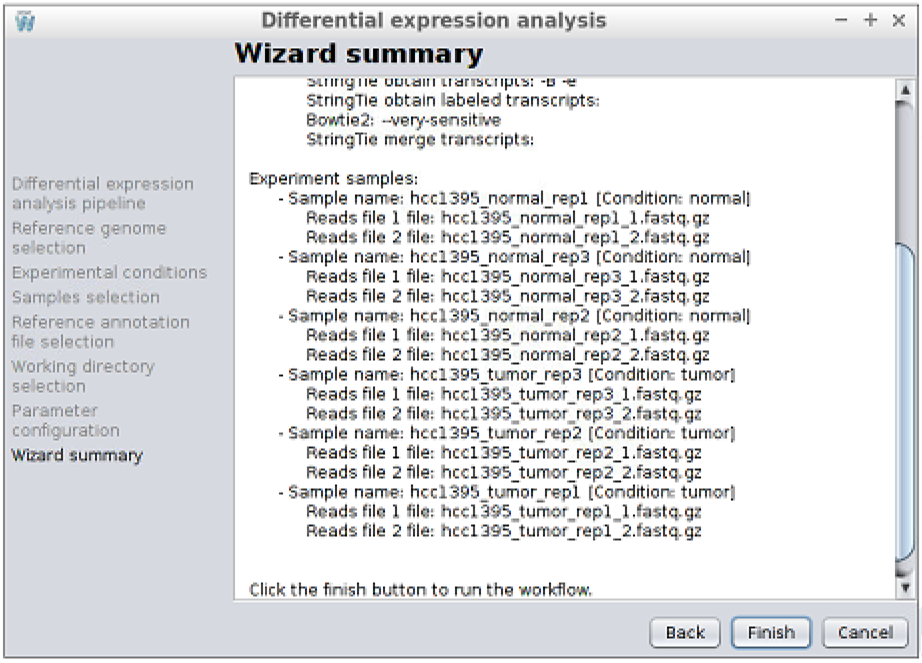
StringTie obtain labeled transcripts:

StringTie merge transcripts:

StringTie obtain transcripts: -B -e

Step 10: workflow configuration summary

In this last step a summary of the workflow configuration is provided. It must be checked carefully in order to ensure that the right files were selected, namely the samples' conditions. If the workflow is executed, this summary is also stored in the selected working directory as *workflow-description.txt*.



Differential expression analysis
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Back
Finish
Cancel

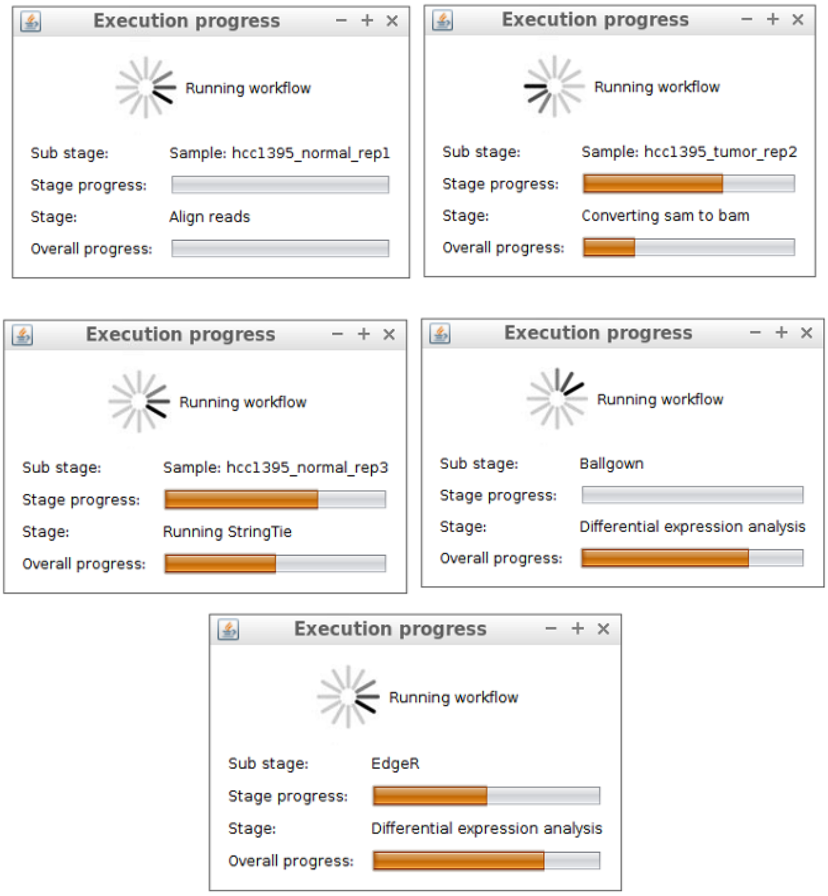
Wizard summary

```
          StringTie obtain transcripts: -B -e
          StringTie obtain labeled transcripts:
          Bowtie2: --very-sensitive
          StringTie merge transcripts:

Experiment samples:
      - Sample name: hcc1395_normal_rep1 [Condition: normal]
          Reads file 1 file: hcc1395_normal_rep1_1.fastq.gz
          Reads file 2 file: hcc1395_normal_rep1_2.fastq.gz
      - Sample name: hcc1395_normal_rep3 [Condition: normal]
          Reads file 1 file: hcc1395_normal_rep3_1.fastq.gz
          Reads file 2 file: hcc1395_normal_rep3_2.fastq.gz
      - Sample name: hcc1395_normal_rep2 [Condition: normal]
          Reads file 1 file: hcc1395_normal_rep2_1.fastq.gz
          Reads file 2 file: hcc1395_normal_rep2_2.fastq.gz
      - Sample name: hcc1395_tumor_rep3 [Condition: tumor]
          Reads file 1 file: hcc1395_tumor_rep3_1.fastq.gz
          Reads file 2 file: hcc1395_tumor_rep3_2.fastq.gz
      - Sample name: hcc1395_tumor_rep2 [Condition: tumor]
          Reads file 1 file: hcc1395_tumor_rep2_1.fastq.gz
          Reads file 2 file: hcc1395_tumor_rep2_2.fastq.gz
      - Sample name: hcc1395_tumor_rep1 [Condition: tumor]
          Reads file 1 file: hcc1395_tumor_rep1_1.fastq.gz
          Reads file 2 file: hcc1395_tumor_rep1_2.fastq.gz


Click the finish button to run the workflow.
```

In order to begin the execution of the workflow, click the *Finish* button.

Step 11: monitoring the workflow execution

While the workflow is being executed, the execution can be monitored in a dialog as the one shown in the following image.



Execution progress

Running workflow

Sub stage: Sample: hcc1395_normal_rep1

Stage progress:

Stage: Align reads

Overall progress:

Execution progress

Running workflow

Sub stage: Sample: hcc1395_tumor_rep2

Stage progress:

Stage: Converting sam to bam

Overall progress:

Execution progress

Running workflow

Sub stage: Sample: hcc1395_normal_rep3

Stage progress:

Stage: Running StringTie

Overall progress:

Execution progress

Running workflow

Sub stage: Ballgown

Stage progress:

Stage: Differential expression analysis

Overall progress:

Execution progress

Running workflow

Sub stage: EdgeR

Stage progress:

Stage: Differential expression analysis

Overall progress:

When the workflow is finished, the Ballgown results are added to the clipboard and it can be explored in the selected working directory.

Step 12: workflow results

The following files and directories are generated by the application in the selected working directory:

- *workflow-description.txt*: a plain text file containing the workflow configuration (input files, experiment description, etc.).
- *run.log*: a log containing the executed commands.
- *read-mapping-statistics.csv*: a table containing the statistics of the alignment for each sample.
- *stringtie*: a directory containing the StringTie merged annotation file.
- Directories for each of the samples: each samples directory contains the files produced by the workflow components such as the alignments (in .sam and .bam formats) and the transcripts calculated by StringTie.
- *analysis:* a directory containing the following subdirectories with the results of each library:

- *analysis/ballgown*: this directory contains the results of the differential expression analysis performed with Ballgown. See subsection *6.1 Ballgown* of section *6. Outputs and visualization* for more information on the generated outputs.
- *analysis/edger*: this directory contains the results of the differential expression analysis performed with edgeR. See subsection *6.2 edgeR* of section *6. Outputs and visualization* for more information on the generated outputs.

4.3 HISAT2, StringTie and R libraries (Ballgown and edgeR)

This workflow was introduced by Pertea, M. et al. [15]. As the title suggests, this workflow makes use of the tools HISAT2 [6] to align sample reads, StringTie [7] to assemble transcripts, and Ballgown and edgeR [10] to perform the differential expression analysis:

- **HISAT2** aligns RNA-Seq reads to a genome and discovers transcript splice sites. This aligner is more exigent than Bowtie2 from a computational point of view but more accurate. Moreover, HISAT2 is superior to Bowtie2 in aligning across intron-sized gaps.
- **StringTie** assembles the alignments into full and partial transcripts, creating multiple isoforms as necessary and estimating the expression levels of all genes and transcripts. StringTie normalises the sequence depth and gene length by reporting the quantification results in FPKM (Fragments Per Kilobase Million) and in TPM (Transcripts Per kilobase Million).
- **Ballgown** takes the transcripts and expression levels from StringTie normalised in FPKM and applies rigorous statistical methods to determine which transcripts are differentially expressed between the conditions. Besides, **edgeR** uses raw count produced by HtSeq and then normalises this raw counts in TMM (Trimmed Mean of M-values).

Here, the workflow is illustrated with an example experiment from chromosome X of *Homo sapiens*. The example data comprises human RNA-Seq samples, but to make this execution faster and simpler for testing, a subset of the reads mapping to human chromosome X has been extracted, which is a relatively gene-rich chromosome that spans 151 megabases, i.e. ~5% of the genome. The conditions to be compared will be females (this being the control) and males (this being the case), and there are six samples for each condition (Table 3), noting that three is the minimum number of replicates for valid statistical results. The RNA-Seq samples will be aligned against the chromosome X of the *H. sapiens* and also annotated against the chromosome X.

Table 3: Population and sex of each sample used in the workflow.

Sample id	Sex	Population
ERR188245	Female	GBR
ERR188428	Female	GBR
ERR188337	Female	GBR

ERR188401	Male	GBR
ERR188257	Male	GBR
ERR188383	Male	GBR
ERR204916	Female	YRI
ERR188234	Female	YRI
ERR188273	Female	YRI
ERR188454	Male	YRI
ERR188104	Male	YRI
ERR188044	Male	YRI

Step 1: download the dataset

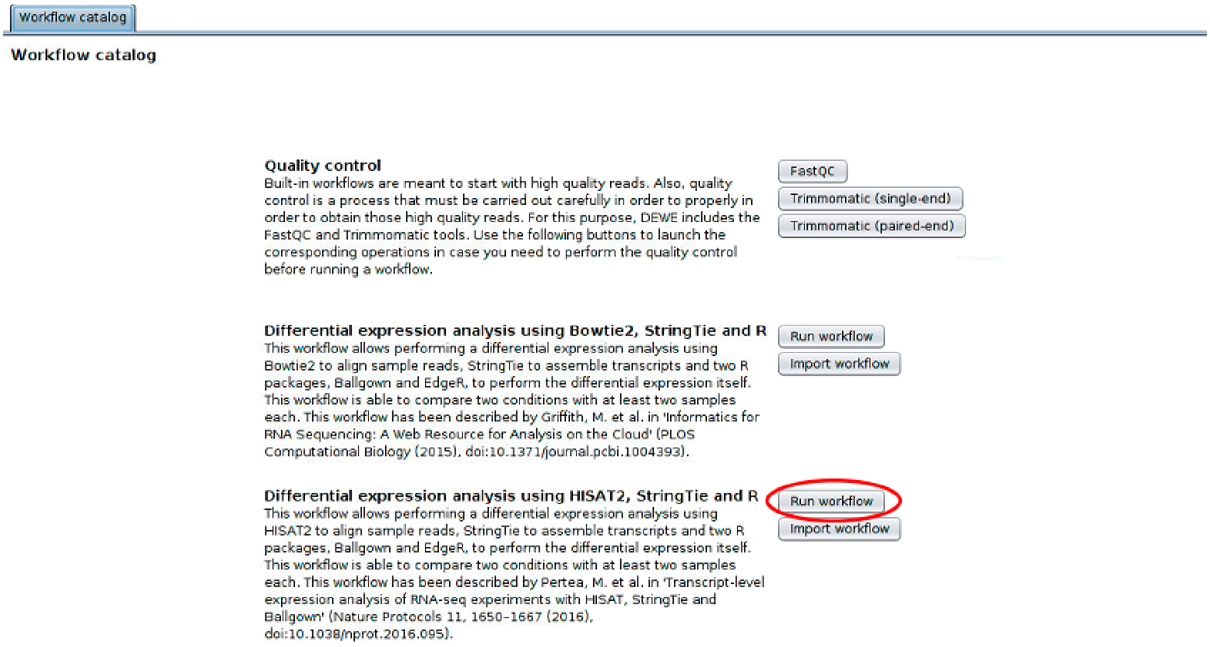
The example dataset is available at the following URL: http://static.sing-group.org/software/DEWE/data/tutorial_data_chrX.zip. This dataset must be downloaded and uncompressed in the application shared folder.

The dataset contains the following files and directories:

- *genes*: a directory containing the reference annotation file called *chrX.gtf*.
- *genome*: a directory containing the reference genome in fasta format.
- *indexes*: a directory containing the HISAT2 indexes of the chromosome X reference genome.
- *samples*: a directory containing two folders:
    - *females*: a directory containing the paired-end reads corresponding to the 6 female samples in the dataset.
    - *males*: a directory containing the paired-end reads corresponding to the 6 male samples in the dataset.
- *geuvadis_phenodata.csv*: a CSV file that contains the phenotype or condition of each sample. For this tutorial, it is important the classification of samples in *male* and *female*.

Step 2: configure the workflow

The next step consists in configuring the workflow. To do so, go to the *Workflow catalo*g and click the *Run workflow* button next to the *HISAT2, StringTie and R libraries* workflow description.

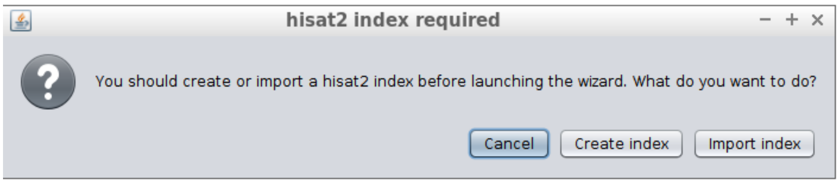


Quality control
Built-in workflows are meant to start with high quality reads. Also, quality control is a process that must be carried out carefully in order to properly in order to obtain those high quality reads. For this purpose, DEWE includes the FastQC and Trimmomatic tools. Use the following buttons to launch the corresponding operations in case you need to perform the quality control before running a workflow.

Differential expression analysis using Bowtie2, StringTie and R
This workflow allows performing a differential expression analysis using Bowtie2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Griffith, M. et al. in 'Informatics for RNA Sequencing: A Web Resource for Analysis on the Cloud' (PLOS Computational Biology (2015). doi:10.1371/journal.pcbi.1004393).

Differential expression analysis using HISAT2, StringTie and R
This workflow allows performing a differential expression analysis using HISAT2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Pertea, M. et al. in 'Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown' (Nature Protocols 11, 1650–1667 (2016), doi:10.1038/nprot.2016.095).

When the workflow is executed for the first time or no HISAT2 reference genome indexes are available, the tool requires the importation or creation of a reference genome index using HISAT2.



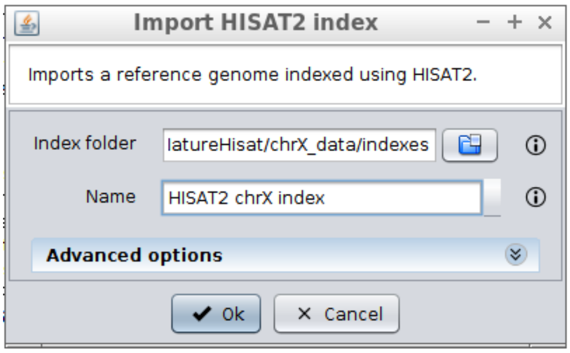
You should create or import a hisat2 index before launching the wizard. What do you want to do?

To build a new index, click the *Create index* button and proceed as explained in section 5.2.1.2. If the reference genome index already exists, as provided in this case study, click the *Import index* button.

Step 3: import the reference genome index

After clicking the *Import index* button, the following dialog will appear, allowing to select the downloaded reference genome index. The following data will be requested:

- *Index folder*: the directory that contains the HISAT2 genome index. Select the *indexes* folder in the case study data. When selecting the folder, the files it contains will not be displayed.
- *Name*: the name for the genome index in order to identify it later.



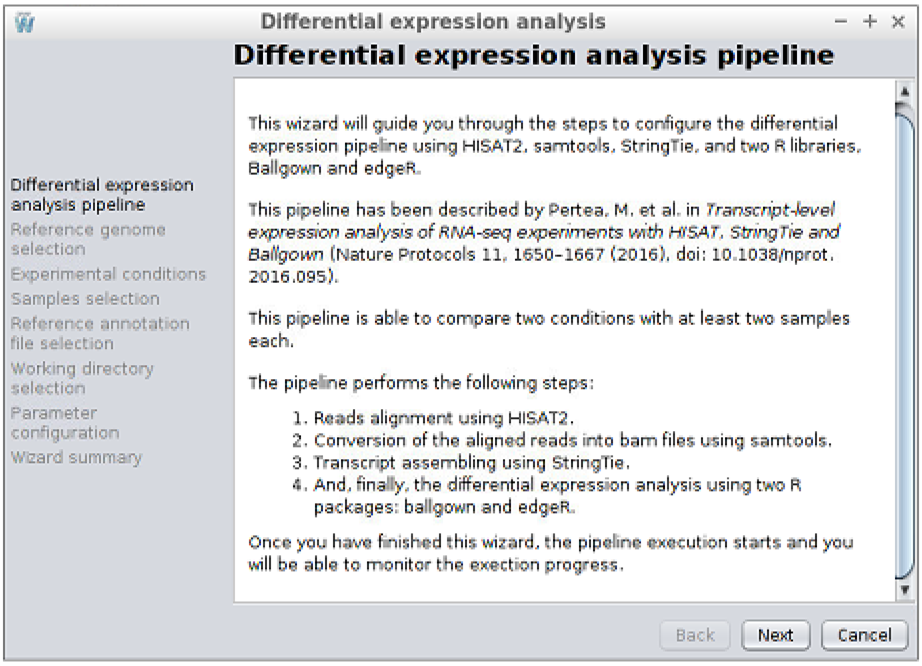
Import HISAT2 index

Imports a reference genome indexed using HISAT2.

Index folder: latureHisat/chrX_data/indexes

Name: HISAT2 chrX index

After setting the required data, click the *Ok* button in order to import the reference genome index. Once the index is successfully imported, it is automatically added to the reference genome indexes database and the workflow configuration assistant opens. The advanced options for import index operation are explained in section 5.1.2.2.



Differential expression analysis

Differential expression analysis pipeline

Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary

This wizard will guide you through the steps to configure the differential expression pipeline using HISAT2, samtools, StringTie, and two R libraries, Ballgown and edgeR.

This pipeline has been described by Pertea, M. et al. in *Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown* (Nature Protocols 11, 1650–1667 (2016), doi: 10.1038/nprot.2016.095).

This pipeline is able to compare two conditions with at least two samples each.

The pipeline performs the following steps:

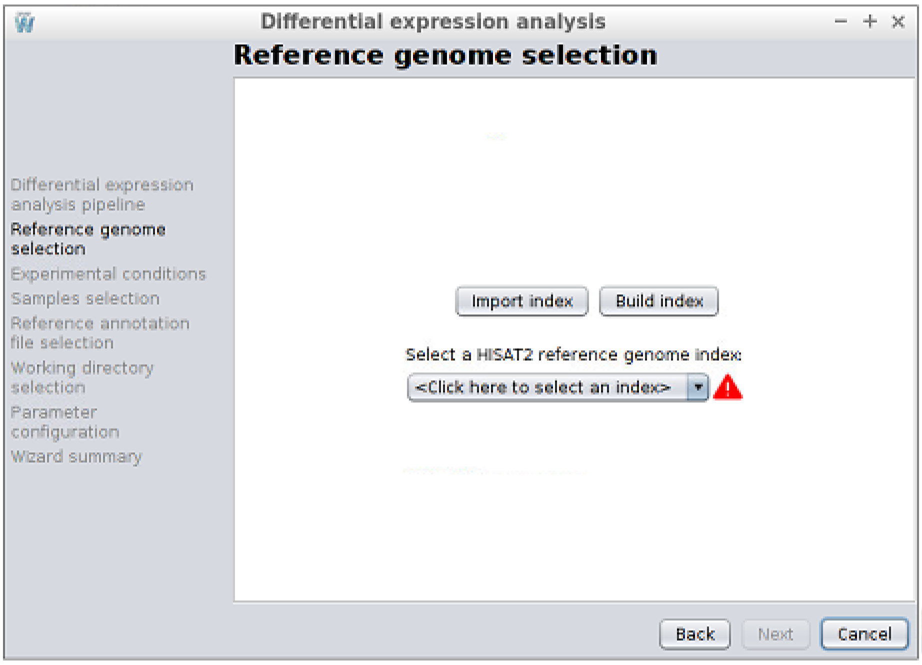
1. Reads alignment using HISAT2.
2. Conversion of the aligned reads into bam files using samtools.
3. Transcript assembling using StringTie.
4. And, finally, the differential expression analysis using two R packages: ballgown and edgeR.

Once you have finished this wizard, the pipeline execution starts and you will be able to monitor the exection progress.

Click the *Next* button to advance to the next step.

Step 4: reference genome selection

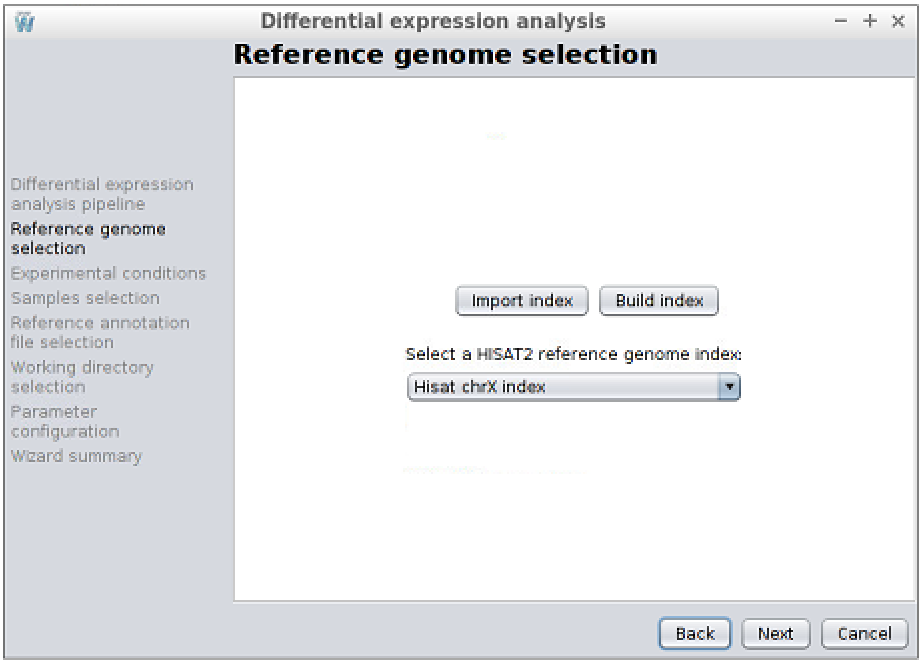
In this step, the reference genome to be used to perform the alignment must be chosen. As shown in the following image, the configuration assistant shows the available HISAT2 indexes. Note that in this step you can also use the *Import index* and *Build index* buttons to import or create a new reference genome index.



Differential expression analysis
Reference genome selection
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Import index
Build index
Back
Next
Cancel

Select a HISAT2 reference genome index: <Click here to select an index>

Select the index imported as shown in the image below and click the *Next* button to advance to the next step.

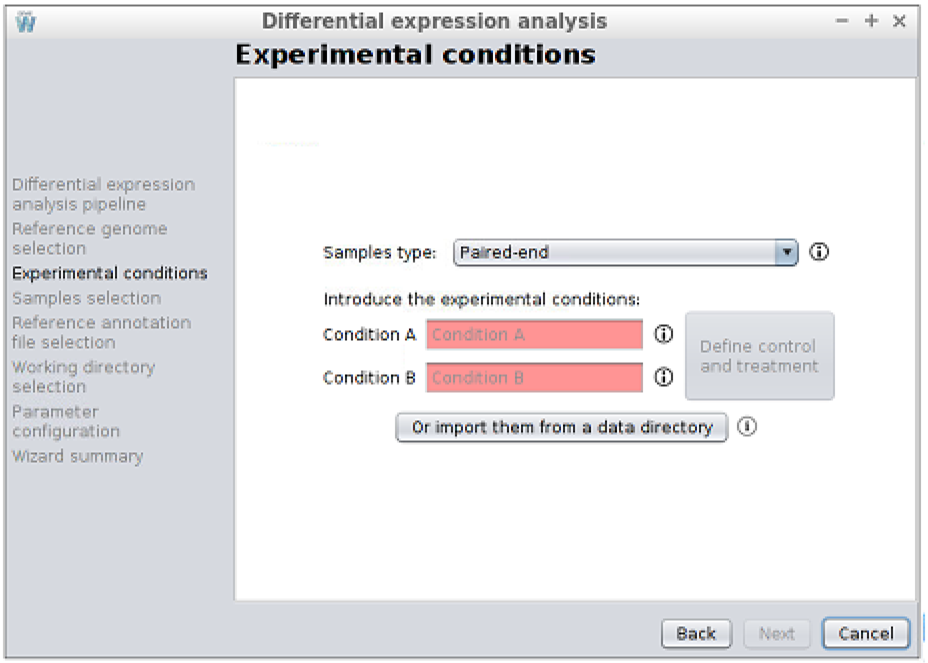


Differential expression analysis
Reference genome selection
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Import index
Build index
Back
Next
Cancel

Select a HISAT2 reference genome index: Hisat chrX index

Step 5: introducing the experimental conditions

In this step, the names of the two experimental conditions of the experiment must be introduced.


Differential expression analysis
Experimental conditions
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Define control and treatment
Or import them from a data directory
Back
Next
Cancel

Samples type: Paired-end

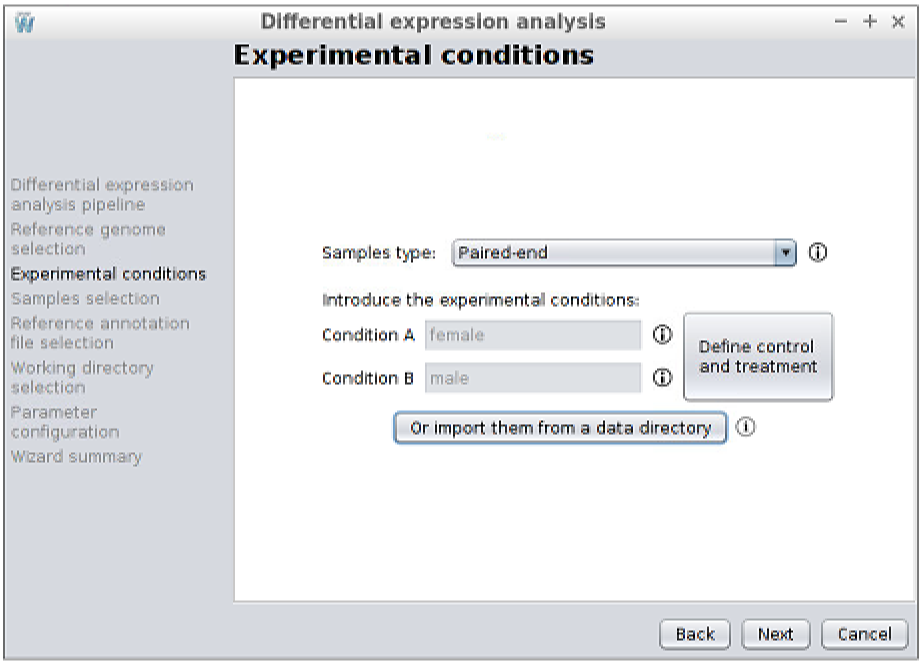
Introduce the experimental conditions:

Condition A: Condition A

Condition B: Condition B

DEWE allows to enter conditions and samples in a simple way by importing them from a directory (by default, paired-end samples type is selected). To do so, click on the "Or import data from a directory" button and select the *samples* directory where the samples are located so that DEWE can automatically import all the data. Note that this directory should be organized as follows:

- It must contain two folders and the name of each folder will correspond to a condition. In this case, the *samples* folder contains the *females* and *males* folders.
- Each of the two folders must contain the pairwise files of the samples and these files must be in .fq, .fastq or .fastq.gz format. In addition, as samples are paired-end, the first reads file must end in _1 and the second reads in _2.


Differential expression analysis
Experimental conditions
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Define control and treatment
Or import them from a data directory
Back
Next
Cancel

Samples type: Paired-end

Introduce the experimental conditions:

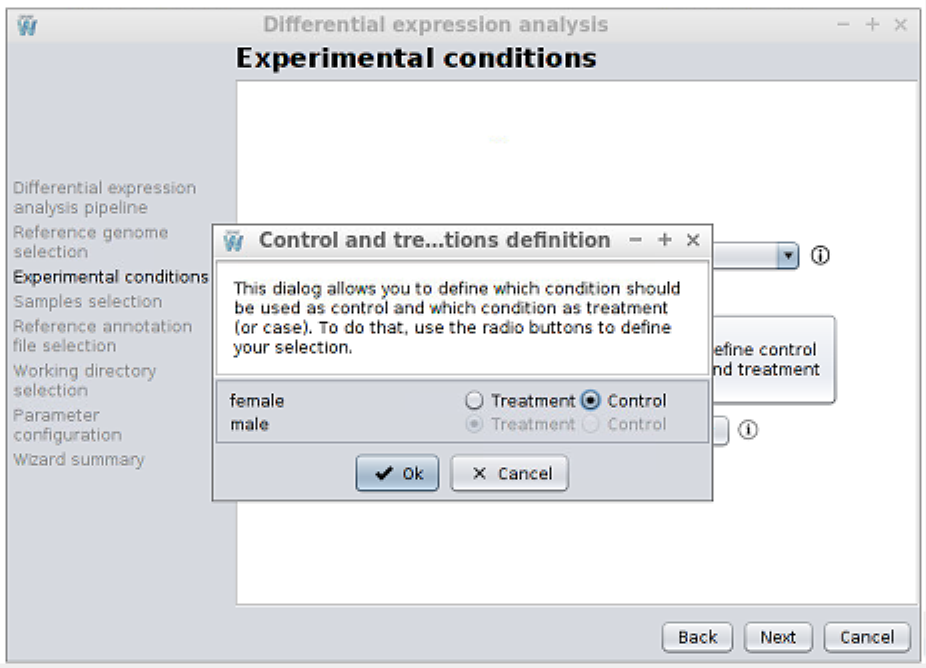
Condition A: female

Condition B: male

To manually enter the data, the *male* and *female* conditions must be specified. The order here is not important, but keep in mind that the program considers as control condition the first condition in alphabetical order.

Advanced: Define control and treatment condition

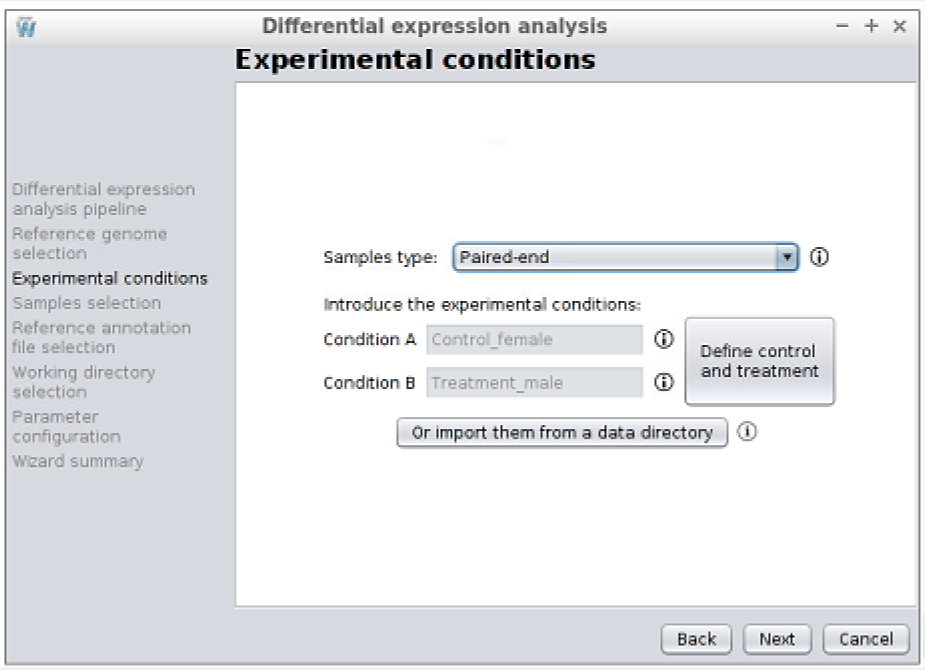
DEWE determines the control condition as the first alphabetical ordered regardless of the order in which they were entered. To change this, there is a button at the right of the conditions text boxes, "Define control and treatment ", which opens a new window where conditions can be defined as Control and Treatment.


Differential expression analysis
Experimental conditions
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Control and tre...tions definition
efine control nd treatment
✓ Ok
✕ Cancel
Back
Next
Cancel

This dialog allows you to define which condition should be used as control and which condition as treatment (or case). To do that, use the radio buttons to define your selection.

female ○ Treatment ● Control
male ● Treatment ○ Control

Once defined and pressed the "Ok" button, DEWE updates the text boxes, with the text “Control_” in front of the control condition name and “Treatment_” in front of the treatment condition name.



Differential expression analysis
Experimental conditions
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary

Samples type: Paired-end

Introduce the experimental conditions:

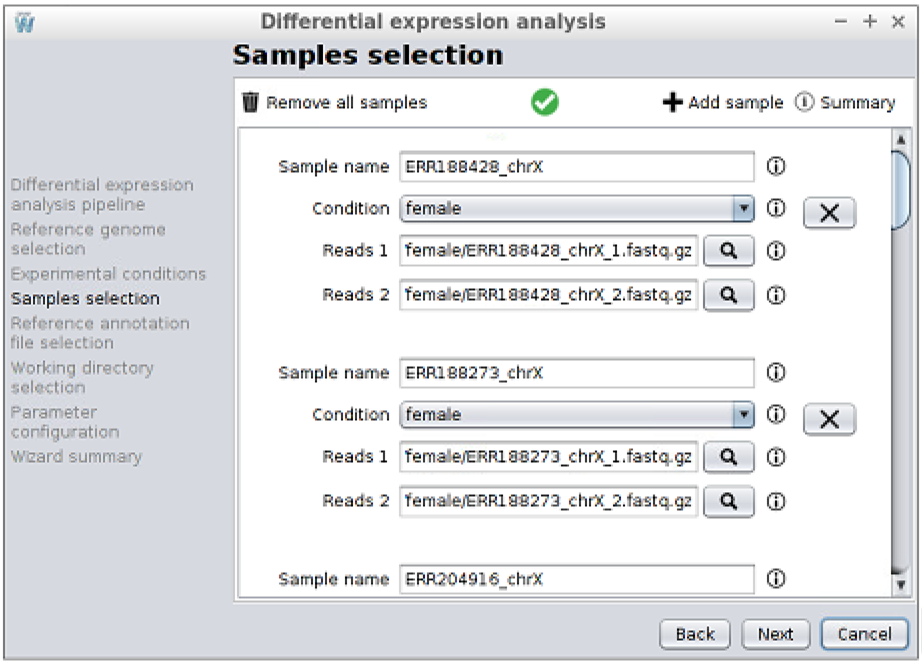
Condition A Control_female

Condition B Treatment_male

Define control and treatment
Or import them from a data directory
Back
Next
Cancel

Step 6 : samples verification *[Optional: samples selection]*

If the samples directory was selected previously, this step can be used to verify that all samples have been selected correctly.



Differential expression analysis
Samples selection
Remove all samples
Add sample
Summary
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary

Sample name ERR188428_chrX

Condition female

Reads 1 female/ERR188428_chrX_1.fastq.gz

Reads 2 female/ERR188428_chrX_2.fastq.gz

Sample name ERR188273_chrX

Condition female

Reads 1 female/ERR188273_chrX_1.fastq.gz

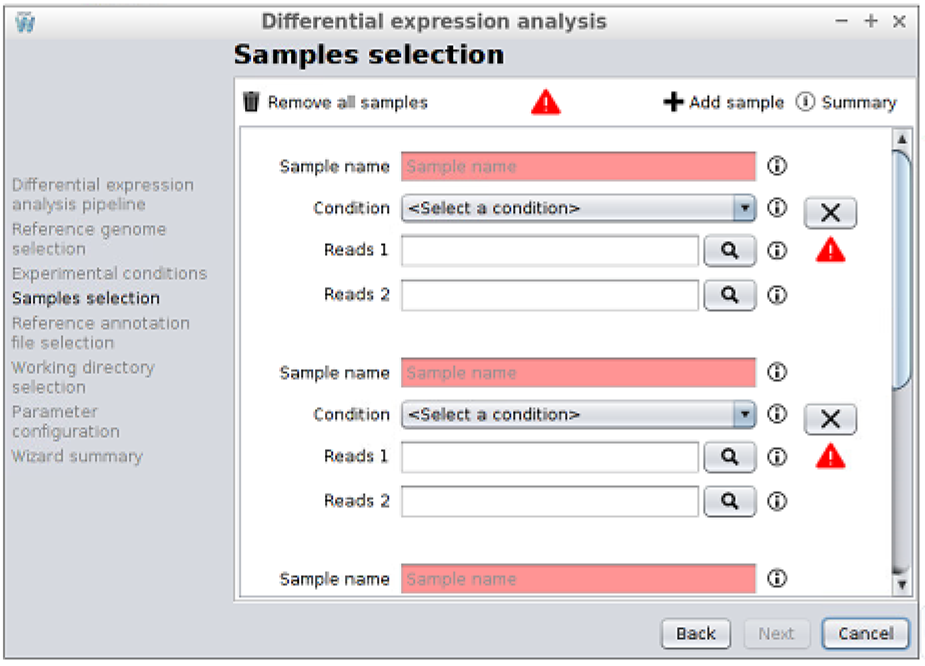
Reads 2 female/ERR188273_chrX_2.fastq.gz

Sample name ERR204916_chrX

Check if all samples are correctly imported and click the *Next* button to advance to the next step.

Optional: Samples selection

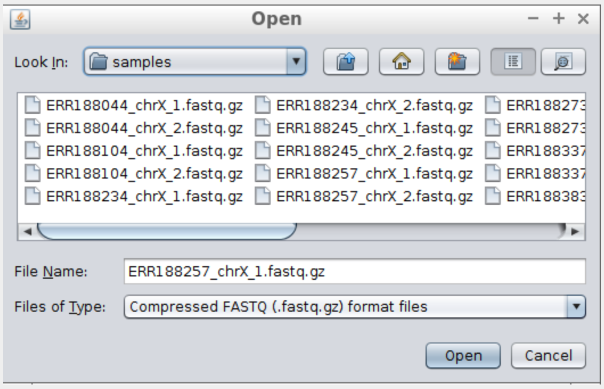
If a samples directory has not been selected in the previous step, the samples must be entered manually (DEWE allows files in .fq, .fastq or .fastq.gz format):



Sample name: Sample name
Condition: <Select a condition>

In order to introduce the samples for the tutorial dataset, follow the next steps:

3. Select the first reads file of each sample (those ending with *_1.fastq.gz* in the *samples* directory) by clicking the 🔍 button.
   - Reads are provided in compressed FASTQ format so if the file browser is empty when browsing into the *samples* directory, then the appropriate file filter must be select as the image below shows.

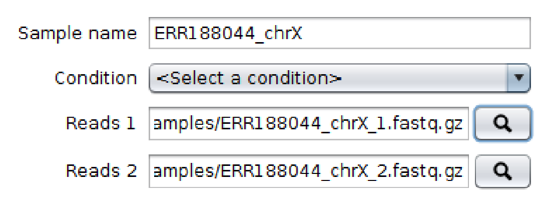


Look In: samples

ERR188044_chrX_1.fastq.gz
ERR188044_chrX_2.fastq.gz
ERR188104_chrX_1.fastq.gz
ERR188104_chrX_2.fastq.gz
ERR188234_chrX_1.fastq.gz
ERR188234_chrX_2.fastq.gz
ERR188245_chrX_1.fastq.gz
ERR188245_chrX_2.fastq.gz
ERR188257_chrX_1.fastq.gz
ERR188257_chrX_2.fastq.gz
ERR18827
ERR18827
ERR18833
ERR18833
ERR18838

File Name: ERR188257_chrX_1.fastq.gz
Files of Type: Compressed FASTQ (.fastq.gz) format files

   - As can be seen in the image below, after selecting the first reads file the second reads file and the sample name are automatically filled in.

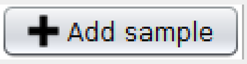


Sample name ERR188044_chrX

Condition <Select a condition>

Reads 1 amples/ERR188044_chrX_1.fastq.gz

Reads 2 amples/ERR188044_chrX_2.fastq.gz

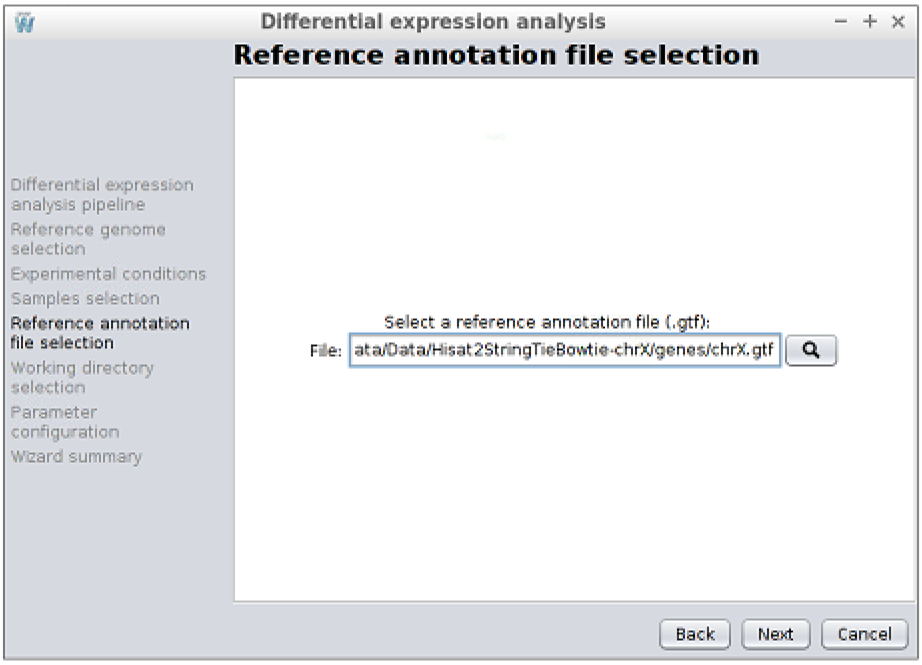
- By default, the list contains four samples. Since in this case you must introduce up to twelve samples, the  button must be clicked to introduce more samples.

4. Assign each of the 12 samples to their corresponding conditions:
    - ERR188044: male.
    - ERR188104: male.
    - ERR188234: female.
    - ERR188245: female.
    - ERR188257: male.
    - ERR188273: female.
    - ERR188337: female.
    - ERR188383: male.
    - ERR188401: male.
    - ERR188428: female.
    - ERR188454: male.
    - ERR204916: female.

After introducing all the samples proceed to the next step. For single-end samples alignment the steps are the same, but only one single-end file is selected for each sample.

Step 7: reference annotation file selection

Select the reference annotation file (GTF format) for the experiment. In this case, the *chrX.gtf* file located in the *genes* directory must be selected. After selecting it, the *Next* button must be clicked to advance to the next step. A ready-to-use reference sequence list and their annotations can be downloaded form the Illumina iGenome site (https://support.illumina.com/sequencing/sequencing_software/igenome.html), taking into account that only eukaryotic organisms can be analysed through DEWE.



Differential expression analysis
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Back
Next
Cancel

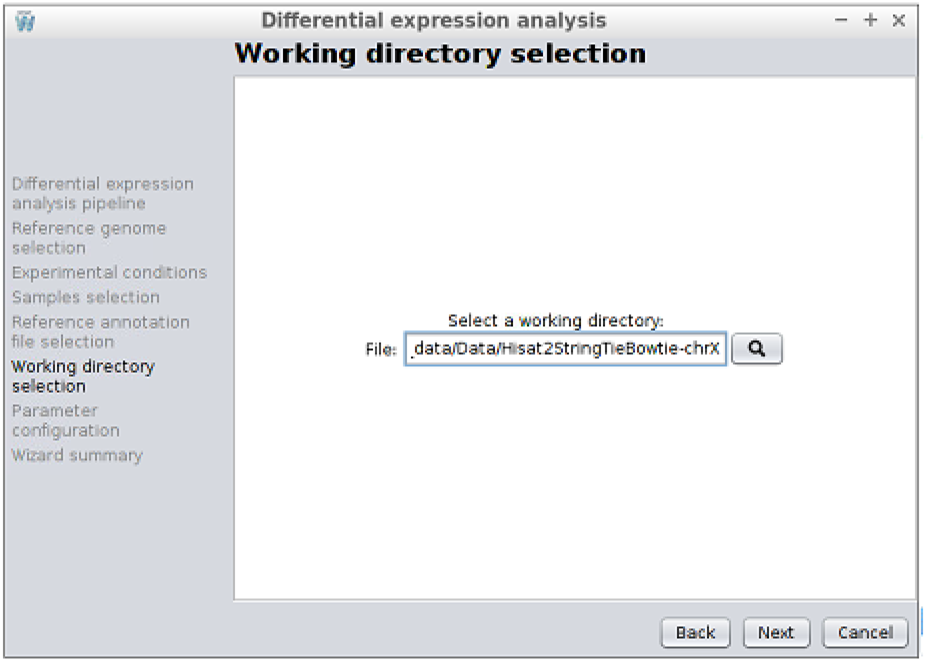
Reference annotation file selection

Select a reference annotation file (.gtf):

File: ata/Data/Hisat2StringTieBowtie-chrX/genes/chrX.gtf

Step 8: working directory selection

Choose the working directory, where the analysis results will be stored, and advance to the final step.



Differential expression analysis
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Back
Next
Cancel

Working directory selection

Select a working directory:

File: data/Data/Hisat2StringTieBowtie-chrX

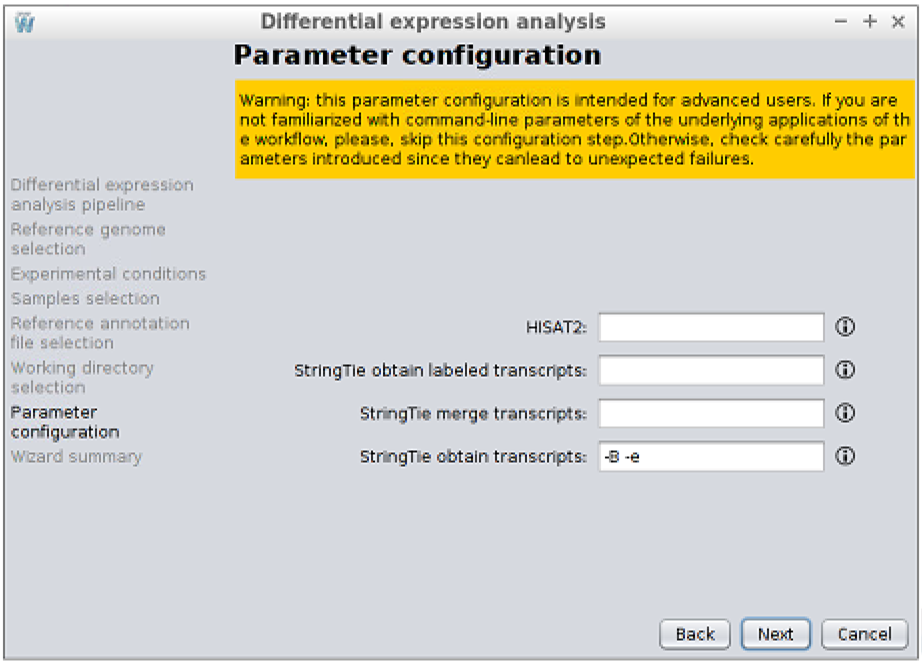
Step 9: parameter configuration

WARNING: this parameter configuration is intended for advanced users. If you are not familiarized with command-line parameters of the underlying applications of the workflow, please, skip this configuration step. Otherwise, check carefully the parameters introduced since they can lead to unexpected failures.

DEWE allows the user to manually introduce additional parameters to the alignment and transcript reconstruction steps, like a command line execution.

- *HISAT2*: custom command-line parameters for the execution of the HISAT2 alignment. For more information on HISAT2 options, please, check the reference manual (https://ccb.jhu.edu/software/hisat2/manual.shtml#command-line-1) and sections 5.3.1.2 and 5.3.2.2.
- *StringTie obtain labeled transcripts*: custom command-line parameters for the execution of StringTie reconstruct labeled transcripts command. For more information on StrinTie options, please, check the reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual) and section 5.5.2 for more details.
- *StringTie merge transcripts*: custom command-line parameters for the execution of the StringTie merge command. For more information on StrinTie options, please, check the reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual) and section 5.5.3 for more details.
- *StringTie obtain transcripts*: custom command-line parameters for the execution of StringTie reconstruct transcripts command. For more information on StrinTie options, please, check the reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual) and section 5.5.1 for more details.

It is important that the parameters are entered correctly or the execution of their respective step will fail.


Differential expression analysis
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary
Back
Next
Cancel

Parameter configuration

Warning: this parameter configuration is intended for advanced users. If you are not familiarized with command-line parameters of the underlying applications of th e workflow, please, skip this configuration step.Otherwise, check carefully the par ameters introduced since they canlead to unexpected failures.

HISAT2:

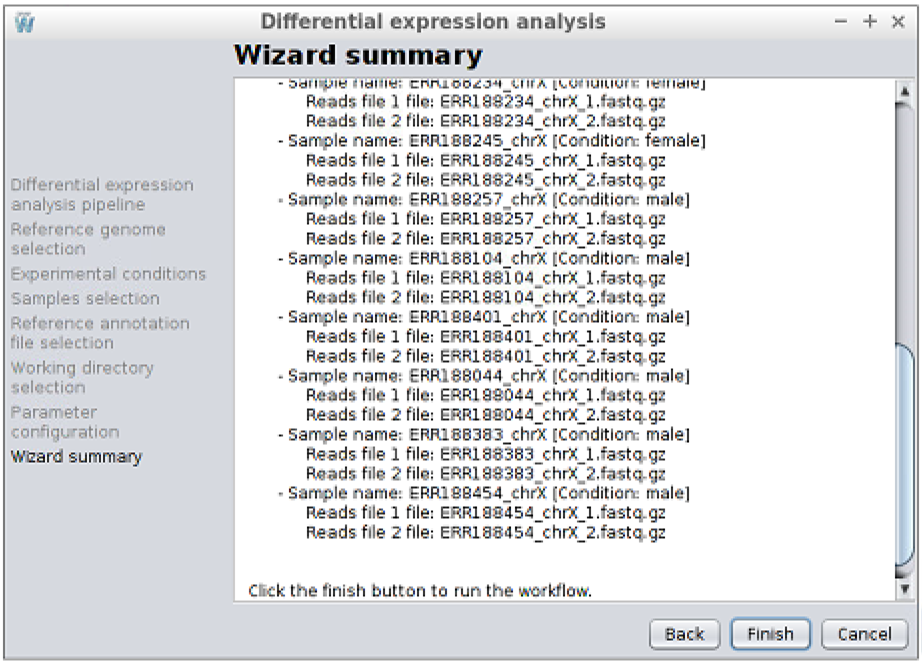
StringTie obtain labeled transcripts:

StringTie merge transcripts:

StringTie obtain transcripts: -B -e

Step 10: workflow configuration summary

In this last step a summary of the workflow configuration is provided. It must be checked carefully in order to ensure that the right files were selected, namely the samples' conditions. If the workflow is executed, this summary is also stored in the selected working directory as *workflow-description.txt*.



Differential expression analysis

Wizard summary

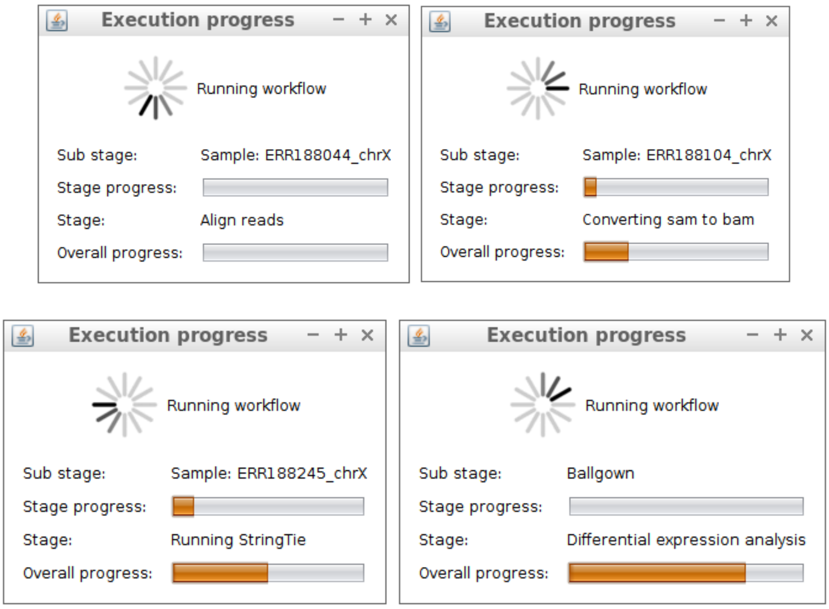
Differential expression analysis pipeline
Reference genome selection
Experimental conditions
Samples selection
Reference annotation file selection
Working directory selection
Parameter configuration
Wizard summary

```
- Sample name: ERR188234_chrX [Condition: female]
    Reads file 1 file: ERR188234_chrX_1.fastq.gz
    Reads file 2 file: ERR188234_chrX_2.fastq.gz
- Sample name: ERR188245_chrX [Condition: female]
    Reads file 1 file: ERR188245_chrX_1.fastq.gz
    Reads file 2 file: ERR188245_chrX_2.fastq.gz
- Sample name: ERR188257_chrX [Condition: male]
    Reads file 1 file: ERR188257_chrX_1.fastq.gz
    Reads file 2 file: ERR188257_chrX_2.fastq.gz
- Sample name: ERR188104_chrX [Condition: male]
    Reads file 1 file: ERR188104_chrX_1.fastq.gz
    Reads file 2 file: ERR188104_chrX_2.fastq.gz
- Sample name: ERR188401_chrX [Condition: male]
    Reads file 1 file: ERR188401_chrX_1.fastq.gz
    Reads file 2 file: ERR188401_chrX_2.fastq.gz
- Sample name: ERR188044_chrX [Condition: male]
    Reads file 1 file: ERR188044_chrX_1.fastq.gz
    Reads file 2 file: ERR188044_chrX_2.fastq.gz
- Sample name: ERR188383_chrX [Condition: male]
    Reads file 1 file: ERR188383_chrX_1.fastq.gz
    Reads file 2 file: ERR188383_chrX_2.fastq.gz
- Sample name: ERR188454_chrX [Condition: male]
    Reads file 1 file: ERR188454_chrX_1.fastq.gz
    Reads file 2 file: ERR188454_chrX_2.fastq.gz

Click the finish button to run the workflow.
```

In order to begin the execution of the workflow, click the *Finish* button.

Step 11: monitoring the workflow execution

While the workflow is being executed, the execution can be monitored in a dialog as the one shown in the following image.



Execution progress

Running workflow

Sub stage: Sample: ERR188044_chrX
Stage progress:
Stage: Align reads
Overall progress:

Execution progress

Running workflow

Sub stage: Sample: ERR188104_chrX
Stage progress:
Stage: Converting sam to bam
Overall progress:

Execution progress

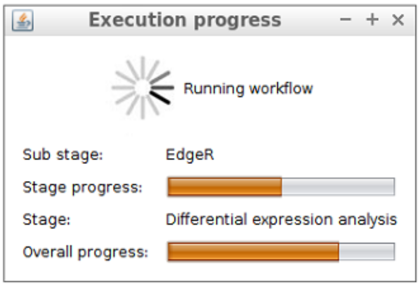
Running workflow

Sub stage: Sample: ERR188245_chrX
Stage progress:
Stage: Running StringTie
Overall progress:

Execution progress

Running workflow

Sub stage: Ballgown
Stage progress:
Stage: Differential expression analysis
Overall progress:



Execution progress

Running workflow

Sub stage: EdgeR

Stage progress:

Stage: Differential expression analysis

Overall progress:

When the workflow is finished, the Ballgown results are added to the clipboard and it can be explored in the selected working directory.

Step 12: workflow results

The following files and directories are generated by the application in the selected working directory:

- *workflow-description.txt*: a plain text file containing the workflow configuration (input files, experiment description, etc.).
- *run.log*: a log containing the executed commands.
- *read-mapping-statistics.csv*: a table containing the statistics of the alignment for each sample.
- *stringtie*: a directory containing the StringTie merged annotation file.
- Directories for each of the samples: each sample's directory contains the files produced by the workflow components such as the alignments (in .sam and .bam formats) and the transcripts calculated by StringTie.
    - *analysis/ballgown*: this directory contains the results of the differential expression analysis performed with Ballgown. See subsection *6.1 Ballgown* of section *6. Outputs and visualization* for more information on the generated outputs.
    - *analysis/edger*: this directory contains the results of the differential expression analysis performed with edgeR. See subsection *6.2 edgeR* of section *6. Outputs and visualization* for more information on the generated outputs.

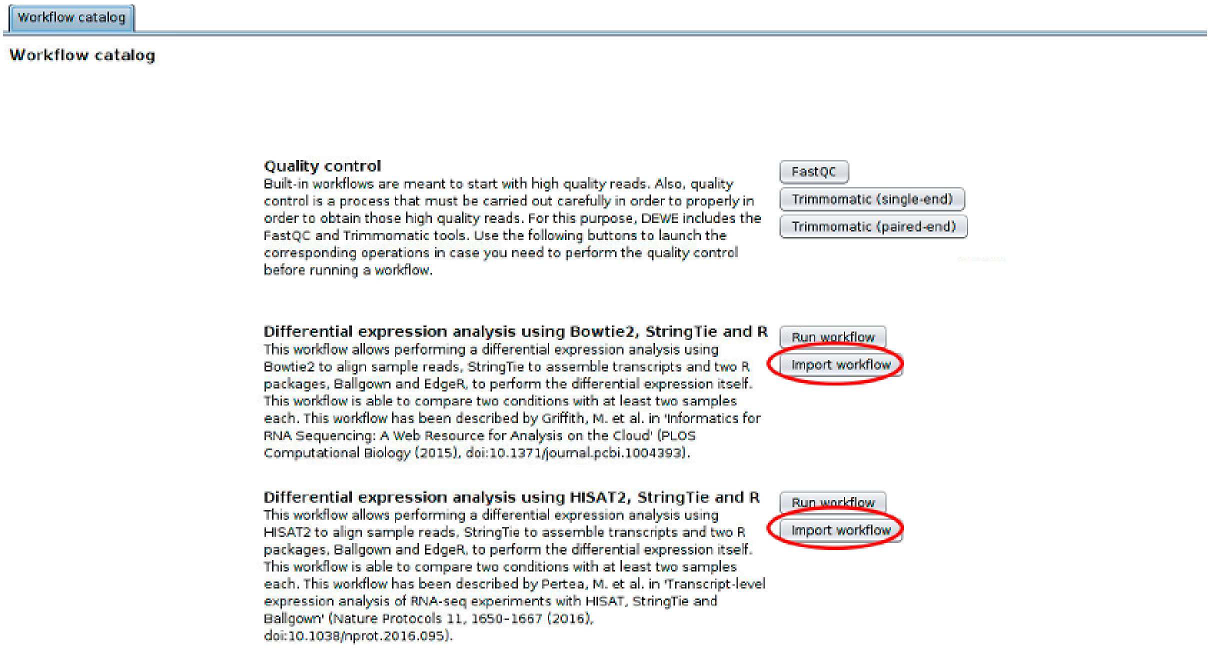
4.3 Configure a workflow using the *workflow.dewe* file

When configuring a workflow, DEWE has two methods. On the one hand, following the steps one by one of sections 4.2 and 4.3 and manually introducing each input. On the other hand, through a workflow configuration file named *workflow.dewe* that automatically introduces all inputs.

This configuration file contains information related to inputs and configuration of the workflow, i.e. the location of the input files (read files of the samples and annotation file), the two conditions, the index selected for the alignment and the output folder. When the configuration file is uploaded, DEWE will fill in all the inputs, but allowing its inspection and modification. A *workflow.dewe* file is generated on the output folder each time an analysis is executed. This file can not be edited manually.

The main advantage of this method is that it allows the re-execution of an analysis quickly. For example, if the user runs an analysis from one workflow and wants to run it in the other, just need to upload the configuration file. Another interesting example would be re-execute an analysis, but changing a single parameter (e.g., remove a sample).

To execute a workflow from a configuration file, in the workflows catalog, click on the "Import workflow" button, which will open a file browser to select the configuration file to be imported.



Workflow catalog

Quality control
Built-in workflows are meant to start with high quality reads. Also, quality control is a process that must be carried out carefully in order to properly in order to obtain those high quality reads. For this purpose, DEWE includes the FastQC and Trimmomatic tools. Use the following buttons to launch the corresponding operations in case you need to perform the quality control before running a workflow.

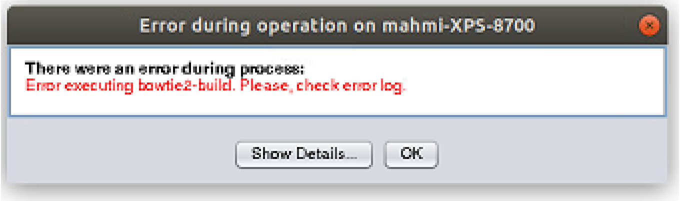
Differential expression analysis using Bowtie2, StringTie and R
This workflow allows performing a differential expression analysis using Bowtie2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Griffith, M. et al. in 'Informatics for RNA Sequencing: A Web Resource for Analysis on the Cloud' (PLOS Computational Biology (2015), doi:10.1371/journal.pcbi.1004393).

Differential expression analysis using HISAT2, StringTie and R
This workflow allows performing a differential expression analysis using HISAT2 to align sample reads, StringTie to assemble transcripts and two R packages, Ballgown and EdgeR, to perform the differential expression itself. This workflow is able to compare two conditions with at least two samples each. This workflow has been described by Pertea, M. et al. in 'Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown' (Nature Protocols 11, 1650-1667 (2016), doi:10.1038/nprot.2016.095).

Once the configuration file has been imported into DEWE, the same steps as in sections 4.2 and 4.3 will be shown, but with all configurations and inputs covered. The rest of the execution is the same as in the sections 4.2 and 4.3.

4.4 Workflow execution FAQ

4.4.1 Error executing bowtie2-build

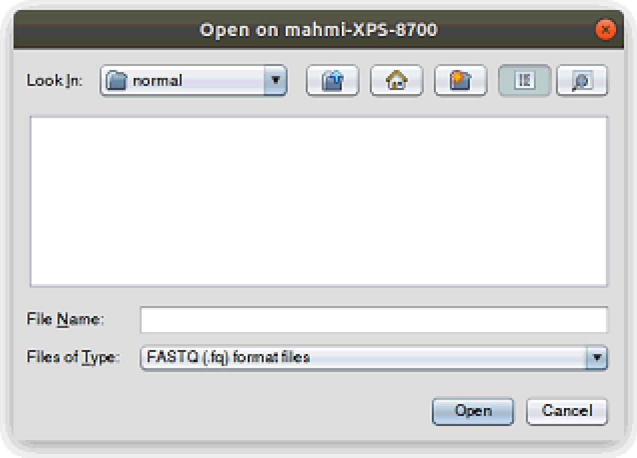


There were an error during process:
Error executing bowtie2-build. Please, check error log.

When DEWE returns this error during index construction, in most cases it means that the genome file used to build the index contains some errors. The user must verify the genome file for errors.

4.4.2 No reads files are displayed in the sample selection

When selecting the sample files manually, the directory where the samples are found may appear empty.

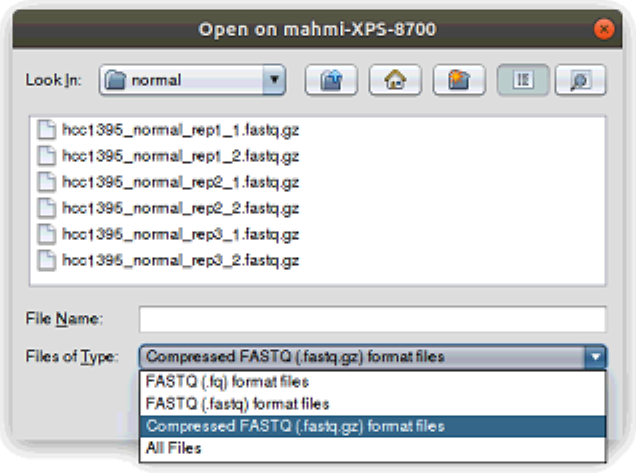


Look In: normal

File Name:

Files of Type: FASTQ (.fq) format files

This happens because by default DEWE searches for files with .fq extension. The user must select in the drop-down menu the correct extension of the reads files so that DEWE recognizes them.


Open on mahmi-XPS-8700
FASTQ (.fq) format files
FASTQ (.fastq) format files
Compressed FASTQ (.fastq.gz) format files
All Files

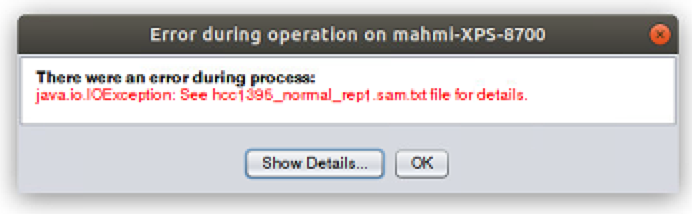
Look In: normal

hcc1395_normal_rep1_1.fastq.gz
hcc1395_normal_rep1_2.fastq.gz
hcc1395_normal_rep2_1.fastq.gz
hcc1395_normal_rep2_2.fastq.gz
hcc1395_normal_rep3_1.fastq.gz
hcc1395_normal_rep3_2.fastq.gz

File Name:

Files of Type: Compressed FASTQ (.fastq.gz) format files

4.4.3 java.io.IOException error during alignment



There were an error during process:
java.io.IOException: See hcc1395_normal_rep1.sam.txt file for details.

When DEWE returns this error during alignment process, this can happen for two reasons. To know the reason, the *sample*.sam.txt file should be inspected to determine the error, in this example *hcc1396_normal_rep1.sam.txt*.

4.2.3.1 Could not locate a Bowtie/HISAT index

```
Could not locate a [Bowtie/HISAT] index corresponding to basename "/mnt/shared/mahmi/rna_seq/sf_data/Data/test/indexes/Homo_sapiens.GRCh38.dna_sm.chromosome.22index"
```

In the sample.sam.txt file

This is because the index was imported / built correctly but later its location has been deleted or changed. The user must import / build the index again.

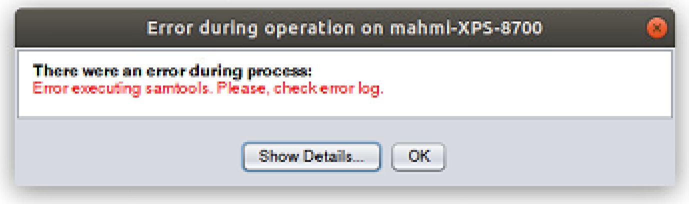
4.2.3.2 Reads file does not look like a FASTQ file

```
Error: reads file does not look like a FASTQ file
```

In the sample.sam.txt file

This is because some of the reading files are wrong. The user must verify the sample files for errors.

4.4.4 Error executing StringTie

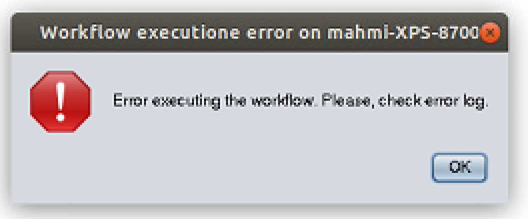


Error during operation on mahmi-XPS-8700

There were an error during process:
Error executing samtools. Please, check error log.

When DEWE returns this error, in most cases it means that the GTF annotation file contains an error. The user must verify the annotation file for errors.

4.4.5 Workflow execution error

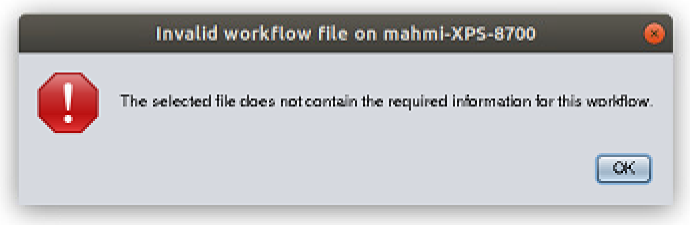


Workflow executione error on mahmi-XPS-8700

Error executing the workflow. Please, check error log.

When DEWE returns this error when importing a workflow, in most cases it means that a *workflow.dewe* configuration file from an old version of DEWE is being imported. The user must manually configure the workflow.

4.4.6 Invalid workflow file



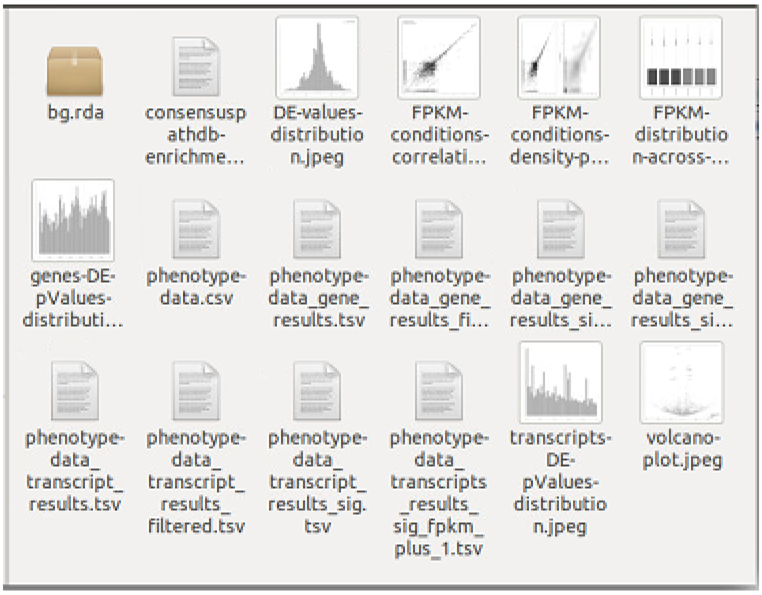
Invalid workflow file on mahmi-XPS-8700

The selected file does not contain the required information for this workflow.

When DEWE returns this error when importing a workflow, in most cases it means that a *workflow.dewe* configuration file from another workflow is being imported. The user must import the file in the correct workflow.

4.4.7 Heatmap and PCA plot have not been generated after Ballgown execution

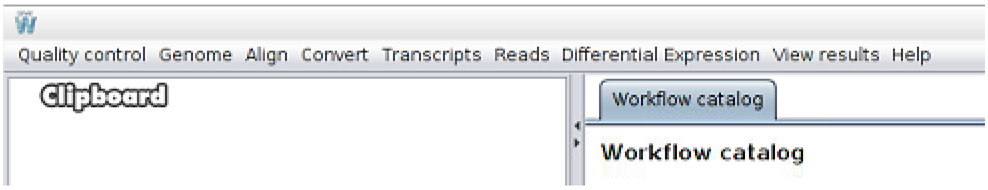
This happens when an analysis does not produce genes with a q-value < 0.05. If in an analysis these figures are not generated, it means that statistically significant genes have not been found.



bg.rda
consensuspathdb-enrichme...
DE-values-distribution.jpeg
FPKM-conditions-correlati...
FPKM-conditions-density-p...
FPKM-distribution-across-...
genes-DE-pValues-distributi...
phenotype-data.csv
phenotype-data_gene_results.tsv
phenotype-data_gene_results_fi...
phenotype-data_gene_results_si...
phenotype-data_gene_results_si...
phenotype-data_transcript_results.tsv
phenotype-data_transcript_results_filtered.tsv
phenotype-data_transcript_results_sig.tsv
phenotype-data_transcripts_results_sig_fpkm_plus_1.tsv
transcripts-DE-pValues-distribution.jpeg
volcano-plot.jpeg

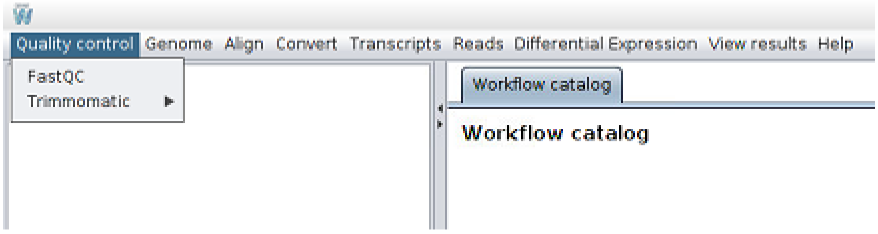
5. Single operations

In addition to being able to run a complete workflow, DEWE also allows the advanced user to execute each step separately. With these single operations, the advanced user is able to execute custom workflows. To do this, in the upper menu the user is provided with all operations.



5.1 The *Quality control* menu

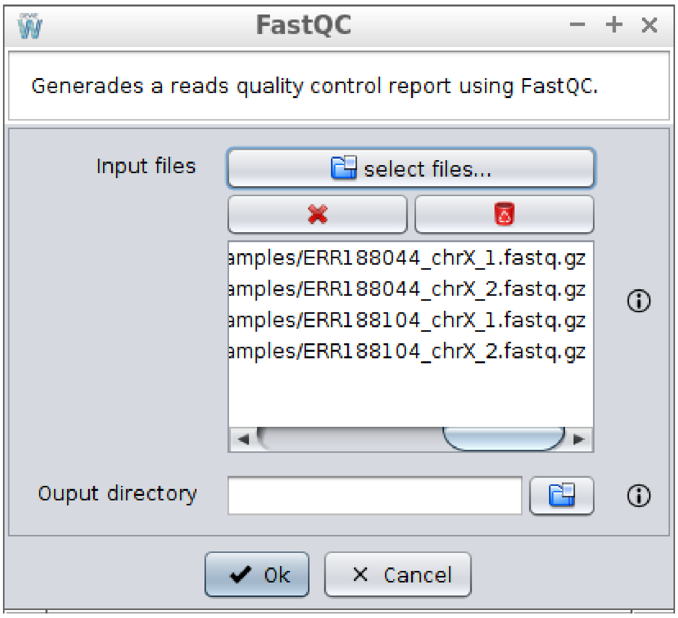
This menu provides operations for performing quality control analysis and reads filtering.



5.1.1 FastQC

This operation allows the generation of a FastQC quality control report for multiple reads files. Clicking on the *Quality control* > *FastQC* button, a new window will be displayed and the following data will be requested:

- *Input files*: the reference annotation file (.gtf).
- *Output directory*: optionally, the directory where the reports must be generated. If not provided, the output report for each reads file is created in the same directory as the reads file being processed.



Generades a reads quality control report using FastQC.

```
amples/ERR188044_chrX_1.fastq.gz
amples/ERR188044_chrX_2.fastq.gz
amples/ERR188104_chrX_1.fastq.gz
amples/ERR188104_chrX_2.fastq.gz
```

Once the *Ok* button is pressed, StringTie analysis starts and a message will be displayed until the end of the process, when an information message is shown.

5.1.2 Trimmomatic

This menu provides operations for performing reads filtering using Trimmomatic.

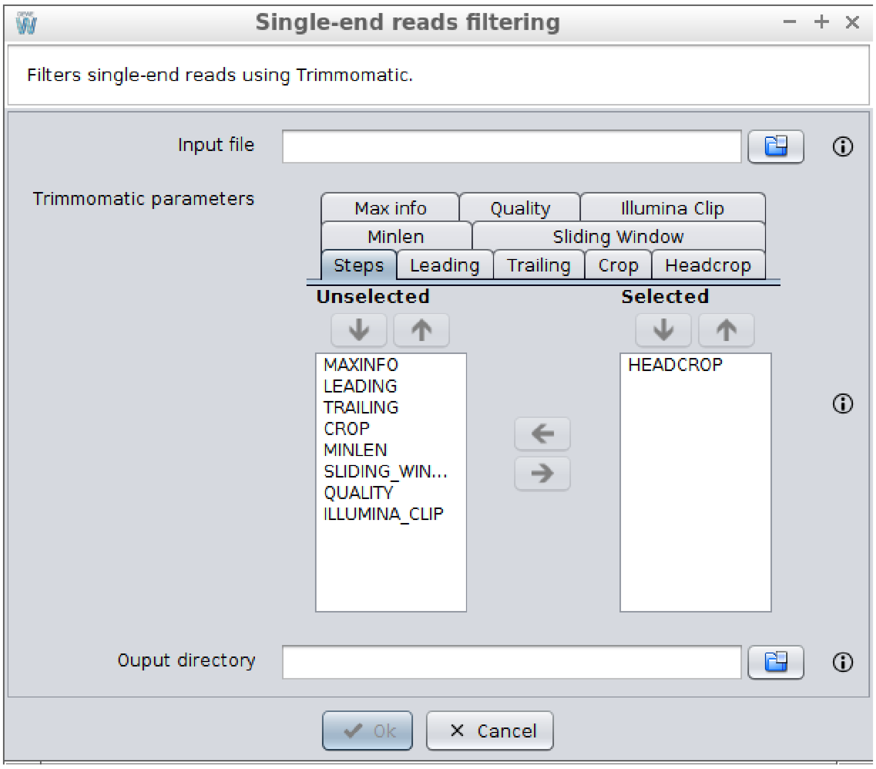
5.1.2.1 Single-end reads filtering

This operation allows filtering single-end raw reads using Trimmomatic. Clicking on the *Quality control* > *Trimmomatic* > *Single-end reads filtering* button, a new window will be displayed and the following data will be requested:

- *Input file*: the input reads file.
- *Trimmomatic parameters*: the steps for trimmomatic and its configuration. The *Steps* tab allows selecting which steps must be applied and define the order in which they should be applied. Then, the other tabs allows configuring each step. The following steps are available:
    - Leading: removes low quality bases from the beginning. As long as a base has a value below this threshold the base is removed and the next base will be investigated.
    - Trailing: removes low quality bases from the end. As long as a base has a value below this threshold the base is removed and the next base (which as trimmomatic is starting from the 3' prime end would be base preceding the just removed base) will be investigated. This approach can be used removing the special illumina 'low quality segment' regions (which are marked with quality score of 2), but we recommend Sliding Window or MaxInfo instead.
    - Crop: removes bases regardless of quality from the end of the read, so that the read has maximally the specified length after this step has been performed. Steps performed after CROP might of course further shorten the read.

- Headcrop: removes the specified number of bases, regardless of quality, from the beginning of the read.
- Minlen: removes reads that fall below the specified minimal length. If required, it should normally be after all other processing steps. Reads removed by this step will be counted and included in the „dropped reads" count presented in the trimmomatic summary.
- Sliding window: performs a sliding window trimming, cutting once the average quality within the window falls below a threshold. By considering multiple bases, a single poor quality base will not cause the removal of high quality data later in the read.
- Max info: performs an adaptive quality trim, balancing the benefits of retaining longer reads against the costs of retaining bases with errors.
- Quality: reencodes the quality part of the FASTQ file to the selected base.
- Illumina clip: finds and removes Illumina adapters.

- *Output directory*: optionally, the directory where the filtered file must be created. If not provided, the output file is created in the same directory as the reads file being filtered.



Single-end reads filtering

Filters single-end reads using Trimmomatic.

Input file

Trimmomatic parameters

Unselected

MAXINFO
LEADING
TRAILING
CROP
MINLEN
SLIDING_WIN...
QUALITY
ILLUMINA_CLIP

Selected

HEADCROP

Ouput directory

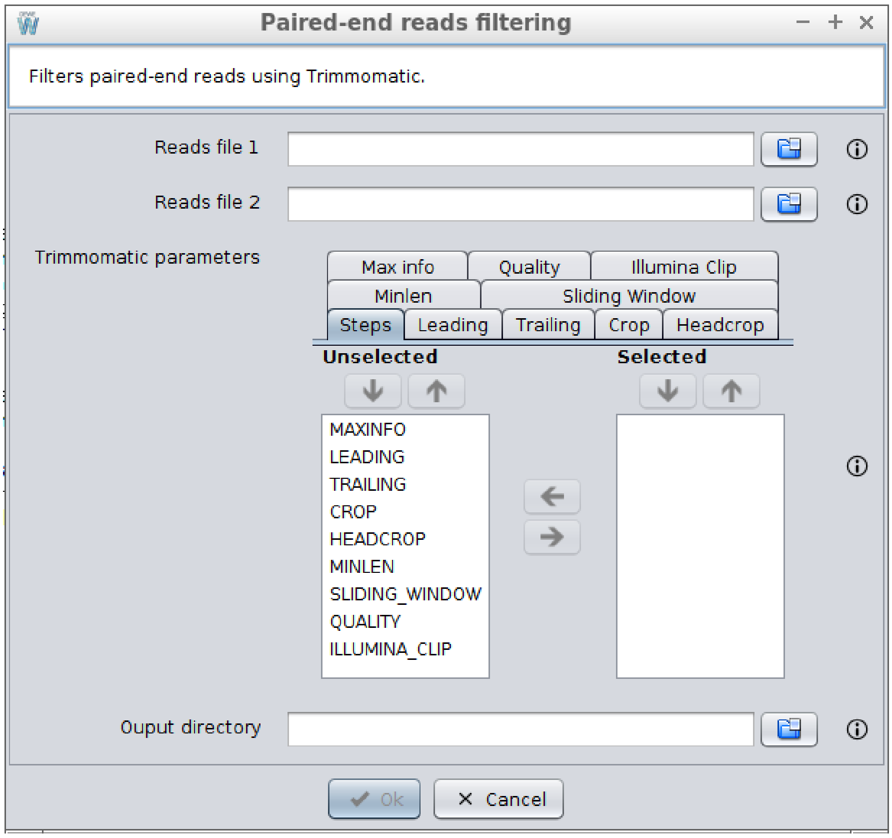
5.1.2.2 Paired-end reads filtering

This operation allows filtering paired-end raw reads using Trimmomatic. Clicking on the *Quality control* > *Trimmomatic* > *Paired-end reads filtering* button, a new window will be displayed and the following data will be requested:

- *Reads file 1*: the first reads file.
- *Reads file 2*: the second reads file
- *Trimmomatic parameters*: the steps for trimmomatic and its configuration. The *Steps* tab allows selecting which steps must be applied and define the order in which they

should be applied. Then, the other tabs allows configuring each step. The following steps are available:

  - Leading: removes low quality bases from the beginning. As long as a base has a value below this threshold the base is removed and the next base will be investigated.
  - Trailing: removes low quality bases from the end. As long as a base has a value below this threshold the base is removed and the next base (which as trimmomatic is starting from the 3' prime end would be base preceding the just removed base) will be investigated. This approach can be used removing the special illumina 'low quality segment' regions (which are marked with quality score of 2), but we recommend Sliding Window or MaxInfo instead.
  - Crop: removes bases regardless of quality from the end of the read, so that the read has maximally the specified length after this step has been performed. Steps performed after CROP might of course further shorten the read.
  - Headcrop: removes the specified number of bases, regardless of quality, from the beginning of the read.
  - Minlen: removes reads that fall below the specified minimal length. If required, it should normally be after all other processing steps. Reads removed by this step will be counted and included in the „dropped reads‟ count presented in the trimmomatic summary.
  - Sliding window: performs a sliding window trimming, cutting once the average quality within the window falls below a threshold. By considering multiple bases, a single poor quality base will not cause the removal of high quality data later in the read.
  - Max info: performs an adaptive quality trim, balancing the benefits of retaining longer reads against the costs of retaining bases with errors.
  - Quality: reencodes the quality part of the FASTQ file to the selected base.
  - Illumina clip: finds and removes Illumina adapters.
- *Output directory*: optionally, the directory where the filtered files must be created. If not provided, the output files are created in the same directory as the reads file being filtered.



Paired-end reads filtering

Filters paired-end reads using Trimmomatic.

Reads file 1

Reads file 2

Trimmomatic parameters

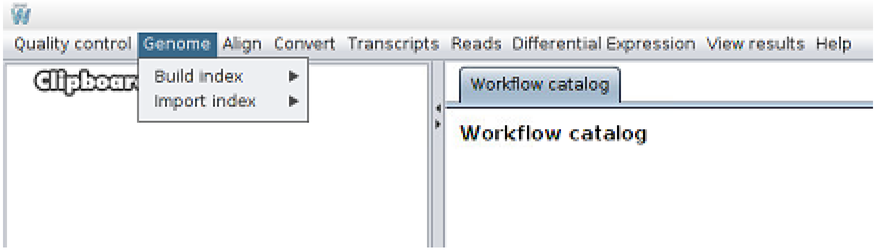
Ouput directory

5.2 The *Genome* menu

To perform the alignment tasks of RNA-Seq samples, it is necessary to have an index against which to align. An index is nothing more than a indexed reference genome against which alignment of the RNA-Seq sequences is performed.

Through this menu, the user can perform all actions related to the management of DEWE indexes. The options available to the user are as follows:

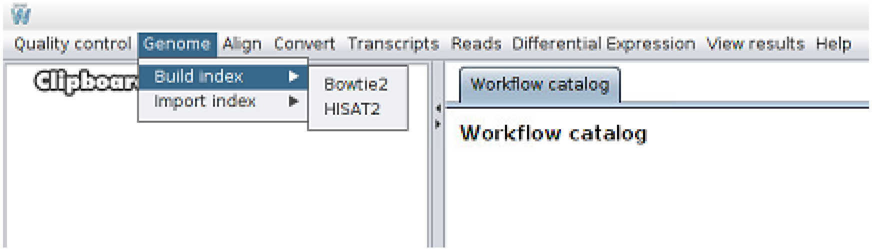
- *Build an index*: construct an index for alignment from a reference genome.
- *Import an index*: import an index for the alignment previously created.



Both operations are available for two alignment tools: Bowtie2 and HISAT2.

5.2.1 Build index

As stated above, an index for Bowtie2 or for HISAT2 can be constructed.

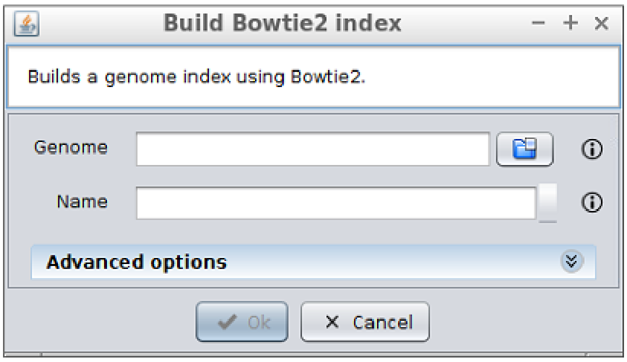


5.2.1.1 Bowtie2

Clicking on the *Genome* > *Build index* > *Bowtie2* button, a new window will be displayed and the following data will be requested:

- *Genome*: the reference genome file for which the index will be created. Must be in .fa format.
- *Name*: the name for the genome index in order to identify it later.

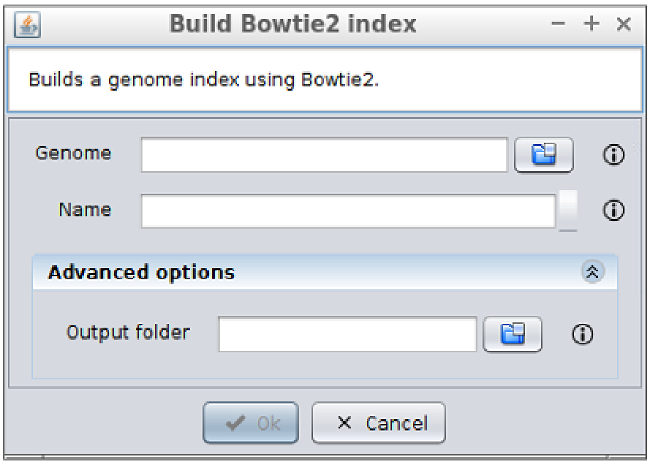
By default, the index is constructed within the folder containing the reference genome.



Build Bowtie2 index

Builds a genome index using Bowtie2.

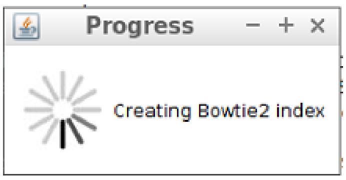
In addition to these data, when building an index with Bowtie2 a menu of *Advanced options* can be displayed. Within this menu there is one more field, *Output folder*, with which the folder where the index will be built can be changed (by default it will be constructed in the folder containing the genome).



Build Bowtie2 index

Builds a genome index using Bowtie2.

Once the *Ok* button is pressed, a message will be displayed until the end of the process.

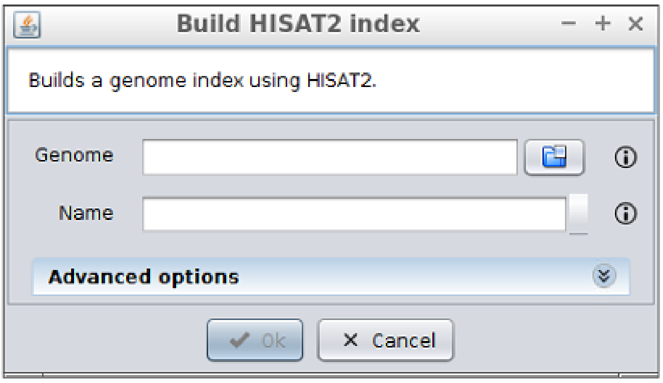

Progress

Creating Bowtie2 index

5.2.1.2 HISAT2

Clicking on the *Genome* > *Build index* > *HISAT2* button, a new window will be displayed and the following data will be requested:

- *Genome*: the reference genome file for which the index will be created. Must be in .fa format.
- *Name*: the name of the genome index.


Build HISAT2 index

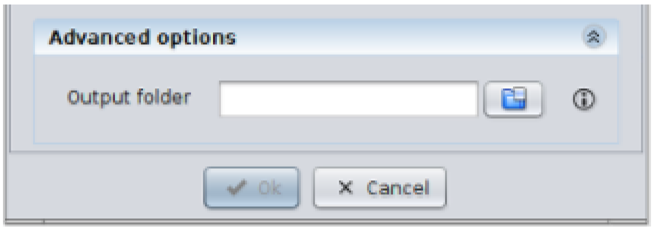
Builds a genome index using HISAT2.

Genome

Name

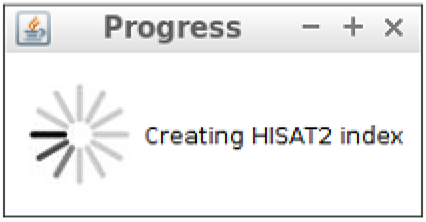
Advanced options

In addition to these data, and similarly to HISAT2's index creation, you may also specify the *Output folder* (in *Advanced options*), i.e. the folder where the index will be built (by default the index will be constructed in the folder containing the genome).


Advanced options

Output folder

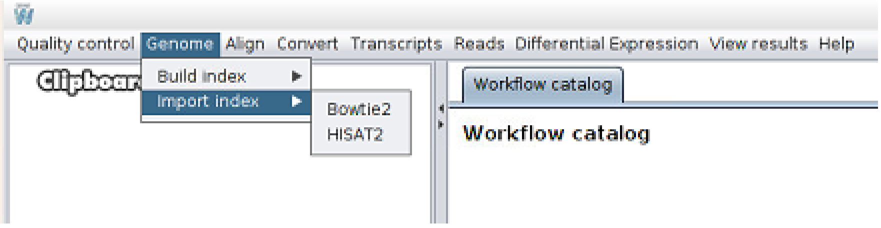
Once the "Ok" button is pressed, a message will be displayed until the end of the process.


Progress

Creating HISAT2 index

5.2.2 Import index

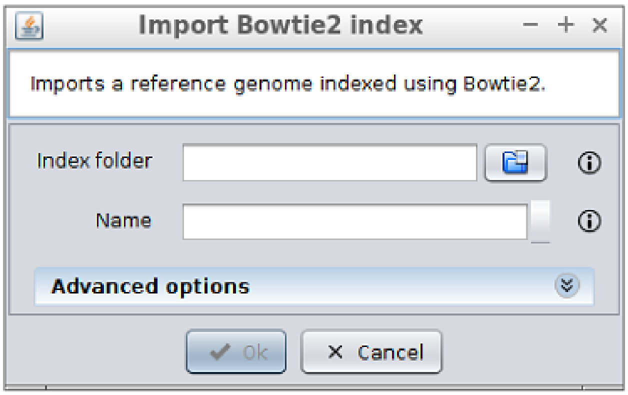
As with index building, you can import an index for both Bowtie2 and HISAT2.



5.2.2.1 Bowtie2

Clicking on the *Genome* > *Import index* > *Bowtie2* button, a new window will be displayed and the following data will be requested:

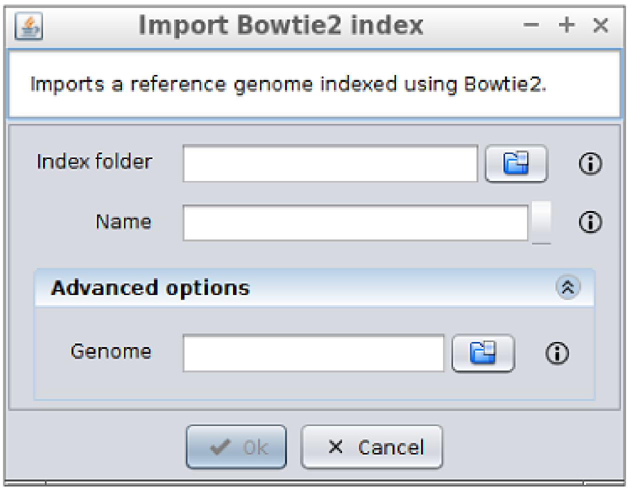
- *Index folder*: the folder that contains the Bowtie2 genome index.
- *Name*: the name for the genome index in order to identify it later.



Import Bowtie2 index

Imports a reference genome indexed using Bowtie2.

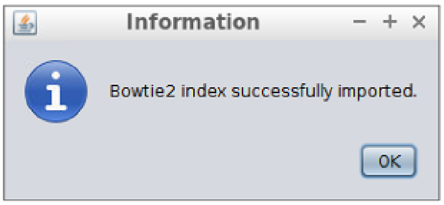
In addition to these data, when importing an Bowtie2 index, a menu of *Advanced options* can be displayed. Within this menu there is one more field, *Genome*, with which the reference genome with which the index was built can be selected (this field is not necessary, but optional)



Import Bowtie2 index

Imports a reference genome indexed using Bowtie2.

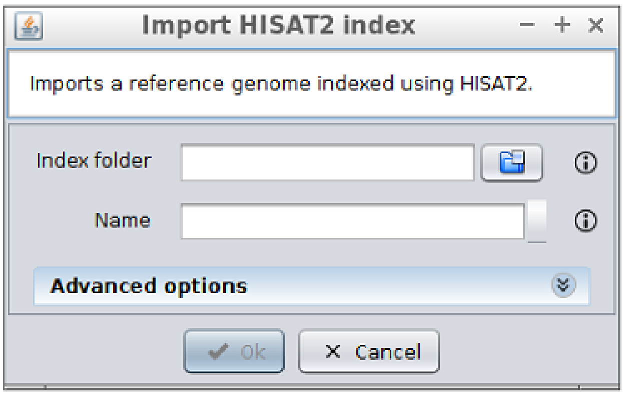
Once the *Ok* button is pressed, if all fields are filled in correctly, the index will be imported.



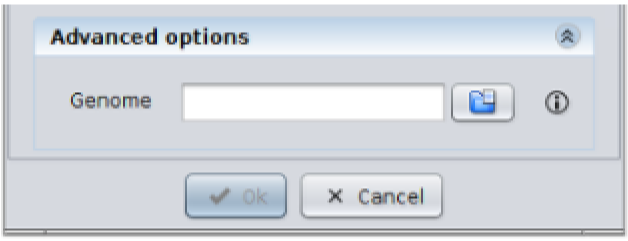
5.2.2.2 HISAT2

Clicking on the *Genome* > *Import index* > *HISAT2* button, a new window will be displayed and the following data will be requested:

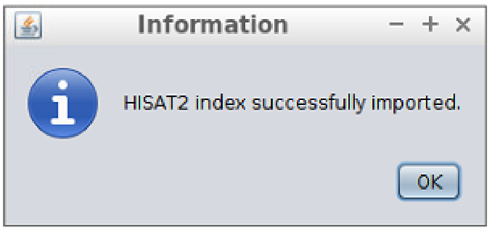
- *Index folder*: the directory that contains the HISAT2 genome index.
- *Name*: the name for the genome index in order to identify it later.



In addition to these data, when importing an HISAT2 index, as with Bowtie2, a menu of *Advanced options* can be displayed. Within this menu there is one more field, *Genome*, with which the reference genome with which the index was built can be selected (this field is not necessary, but optional)

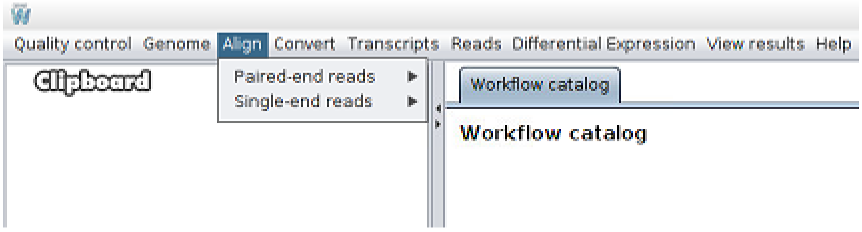


Once the Ok button is pressed, if all fields are filled in correctly, the index will be imported.



5.3 The *Align* menu

This menu provides operations for performing reads alignment using Bowtie2 or HISAT2.

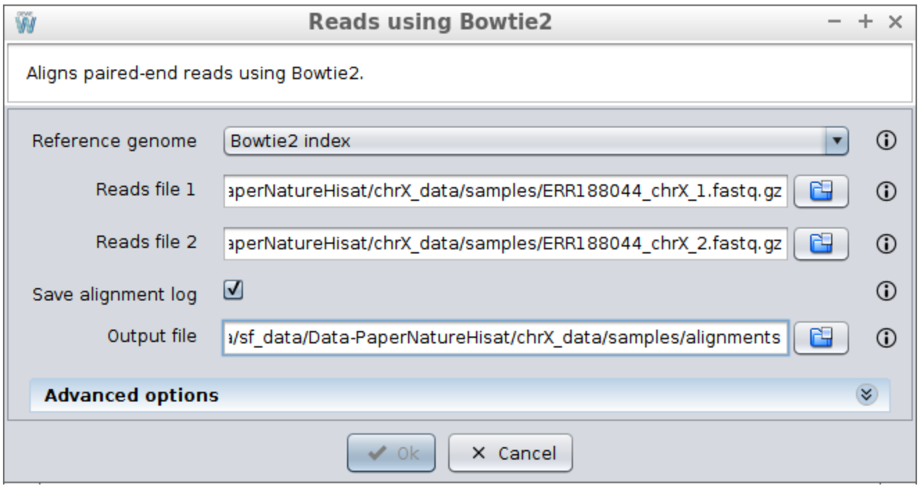


5.3.1 Align paired-end reads

5.3.1.1. Bowtie2

Clicking on the *Align* > *Paired-end Reads* > *Bowtie2* button, a new window will be displayed and the following data will be requested:

- *Reference genome*: the reference genome index to use. It must have been previously built or imported into the application.
- *Reads file 1*: the first reads file. Must be in .fq, .fastq or .fastq.gz format.
- *Reads file 2*: the second reads file. Must be in .fq, .fastq or .fastq.gz format.
- *Save alignment log*: whether the alignment log must be saved or not.
- *Output file*: the output file to save the alignments. Should be a .sam file.


Reads using Bowtie2

Aligns paired-end reads using Bowtie2.

Reference genome: Bowtie2 index

Reads file 1: aperNatureHisat/chrX_data/samples/ERR188044_chrX_1.fastq.gz

Reads file 2: aperNatureHisat/chrX_data/samples/ERR188044_chrX_2.fastq.gz

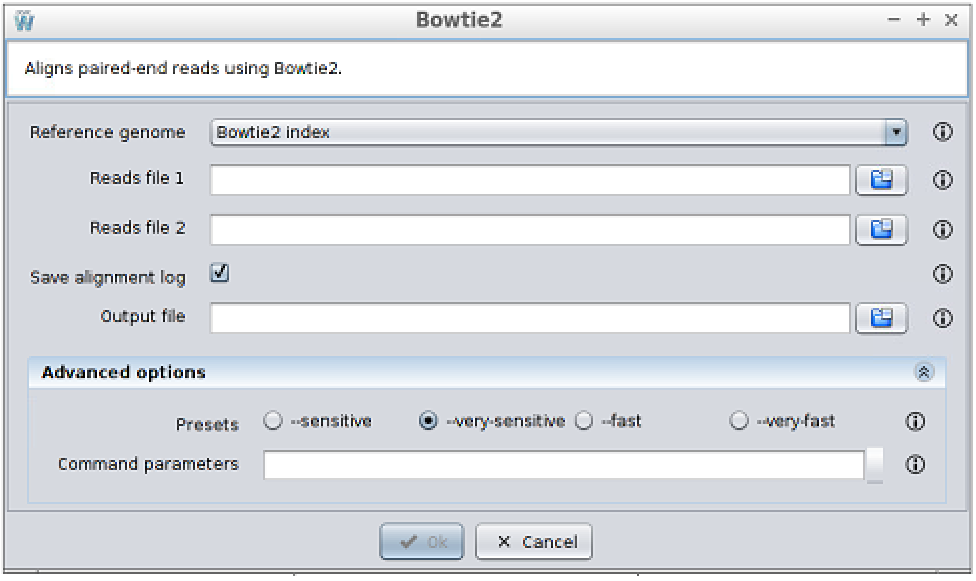
Save alignment log: ☑

Output file: a/sf_data/Data-PaperNatureHisat/chrX_data/samples/alignments

In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is two more parameter:

- *Presets*, that allows the selection of a preset option for the Bowtie2 *--end-to-end* mode. The default preset option is “--very-sensitive”;
- *Command parameters*, that allows the user to manually introduce others Bowtie2 alignment parameters, like a command line execution. These parameters overwrite those defined in *Presets*. It is important that the parameters are entered correctly or the execution of Bowtie2 will fail.

For more information on this advanced options, please, check the Bowtie2 reference manual (http://bowtie-bio.sourceforge.net/bowtie2/manual.shtml#options).



Aligns paired-end reads using Bowtie2.

Reference genome: Bowtie2 index

Reads file 1

Reads file 2

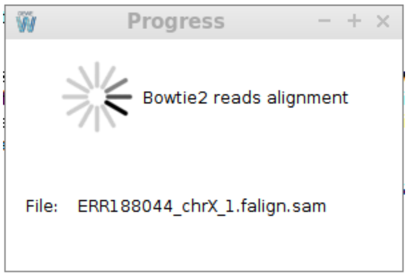
Save alignment log ☑

Output file

Presets ○ --sensitive ● --very-sensitive ○ --fast ○ --very-fast

Command parameters

Once the *Ok* button is pressed, a message will be displayed until the end of the process.



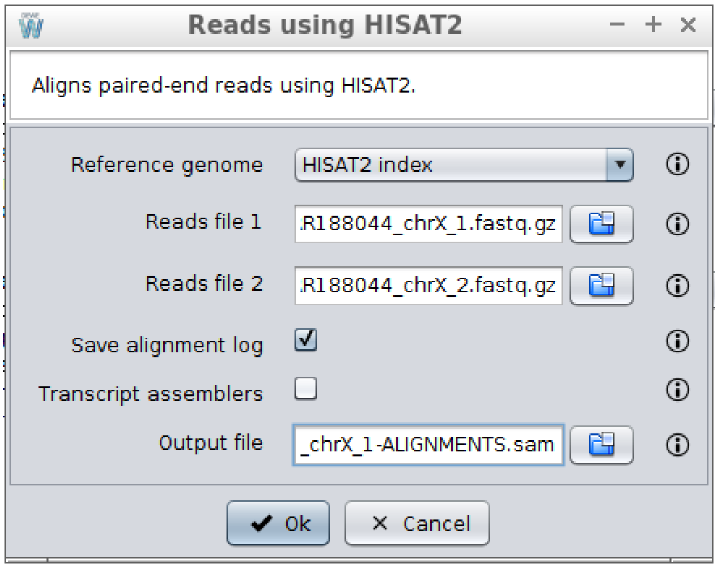
Bowtie2 reads alignment

File: ERR188044_chrX_1.falign.sam

5.3.1.2 HISAT2

Clicking on the *Align* > *Paired-end Reads* > *HISAT2* button, a new window will be displayed and the following data will be requested:

- *Reference genome*: the reference genome index to use. It must have been previously built or imported into the application.
- *Reads file 1*: the first reads file. Must be in .fq, .fastq or .fastq.gz format.
- *Reads file 2*: the second reads file. Must be in .fq, .fastq or .fastq.gz format.
- *Save alignment log*: whether the alignment log must be saved or not.
- *Transcript assemblers*: whether to report alignments tailored for transcript assemblers (including Stringtie) or not. With this option, HISAT2 requires longer anchor lengths for de novo discovery of splice sites. This leads to fewer alignments with short-anchors, which helps transcript assemblers improve significantly in computation and memory usage.
- *Output file*: the output file to save the alignments. Should be a .sam file.


Reads using HISAT2

Aligns paired-end reads using HISAT2.

Reference genome: HISAT2 index

Reads file 1: .R188044_chrX_1.fastq.gz

Reads file 2: .R188044_chrX_2.fastq.gz

Save alignment log: ☑

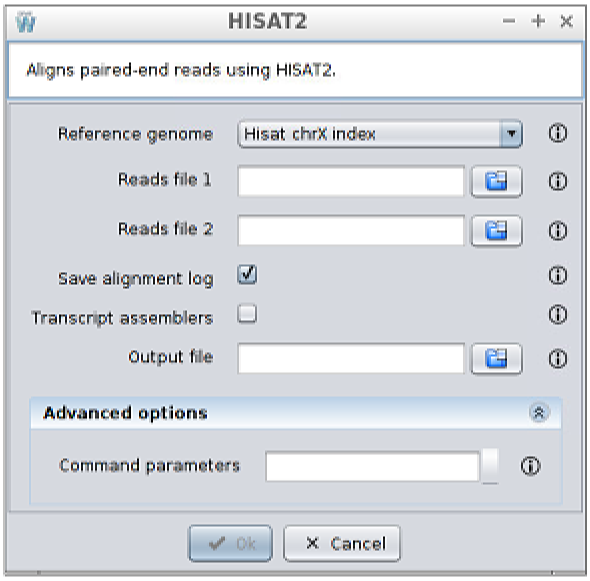
Transcript assemblers: □

Output file: _chrX_1-ALIGNMENTS.sam

In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is one more parameter:

- *Command parameters*, that allows the user to manually introduce others HISAT2 alignment parameters, like a command line execution. It is important that the parameters are entered correctly or the execution of HISAT2 will fail.

For more information on this advanced options, please, check the HISAT2 reference manual (https://ccb.jhu.edu/software/hisat2/manual.shtml#command-line-1).


HISAT2

Aligns paired-end reads using HISAT2.

Reference genome: Hisat chrX index

Reads file 1:

Reads file 2:

Save alignment log: ☑

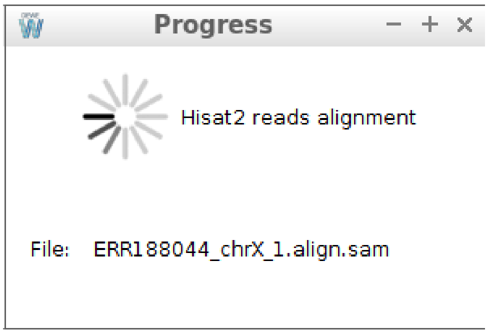
Transcript assemblers: □

Output file:

Advanced options

Command parameters:

Once the *Ok* button is pressed, a message will be displayed until the end of the process.



Progress

Hisat2 reads alignment

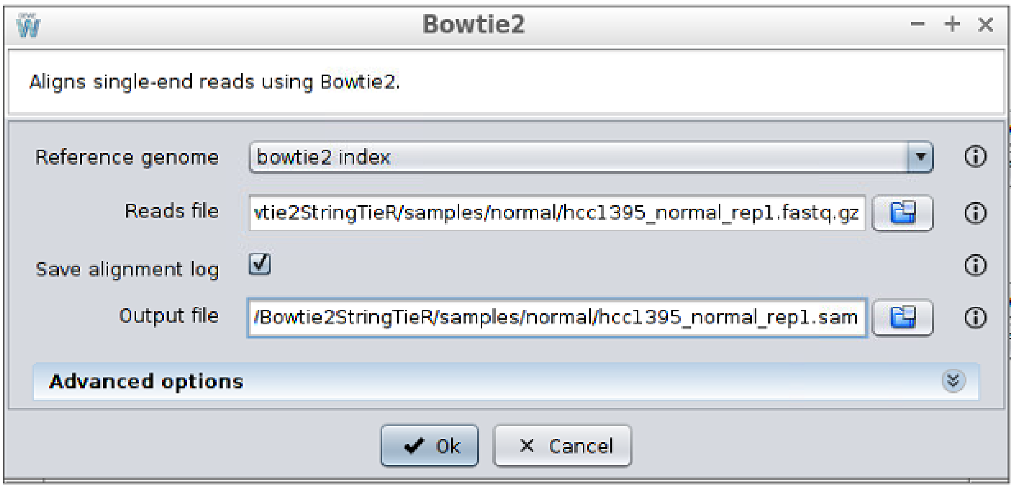
File: ERR188044_chrX_1.align.sam

5.3.2 Align single-end reads

5.3.2.1. Bowtie2

Clicking on the *Align* > *Single-end Reads* > *Bowtie2* button, a new window will be displayed and the following data will be requested:

- *Reference genome*: the reference genome index to use. It must have been previously built or imported into the application.
- *Reads file 1*: the reads file. Must be in .fq, .fastq or .fastq.gz format.
- *Save alignment log*: whether the alignment log must be saved or not.
- *Output file*: the output file to save the alignments. Should be a .sam file.



Bowtie2

Aligns single-end reads using Bowtie2.

Reference genome: bowtie2 index

Reads file: vtie2StringTieR/samples/normal/hcc1395_normal_rep1.fastq.gz

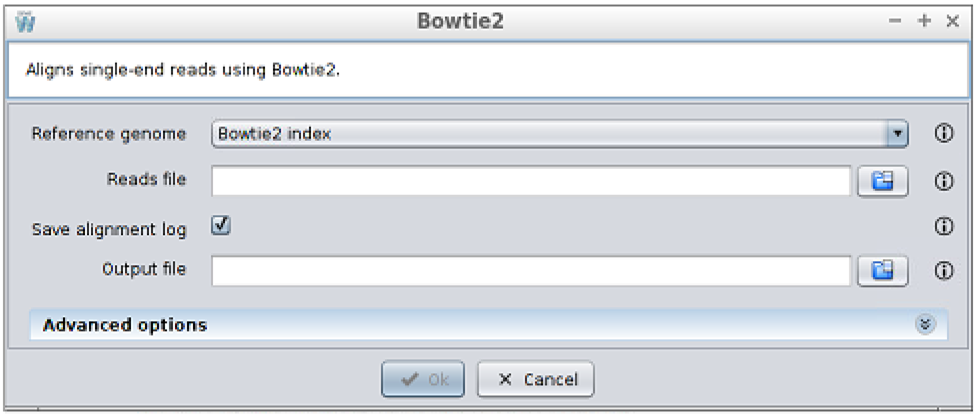
Save alignment log: ☑

Output file: /Bowtie2StringTieR/samples/normal/hcc1395_normal_rep1.sam

In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is two more parameter:

- *Presets*, that allows the selection of a preset option for the Bowtie2 *--end-to-end* mode. The default preset option is "--very-sensitive";
- *Command parameters*, that allows the user to manually introduce others Bowtie2 alignment parameters, like a command line execution. These parameters overwrite those defined in *Presets*. It is important that the parameters are entered correctly or the execution of Bowtie2 will fail.

For more information on this advanced options, please, check the Bowtie2 reference manual (http://bowtie-bio.sourceforge.net/bowtie2/manual.shtml#options).


Bowtie2

Aligns single-end reads using Bowtie2.

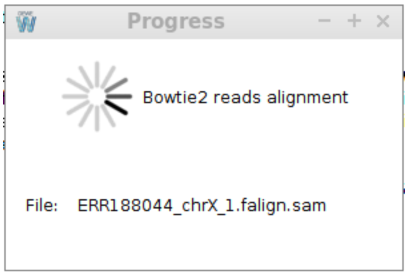
Reference genome: Bowtie2 index

Reads file:

Save alignment log: ☑

Output file:

Once the *Ok* button is pressed, a message will be displayed until the end of the process.


Progress

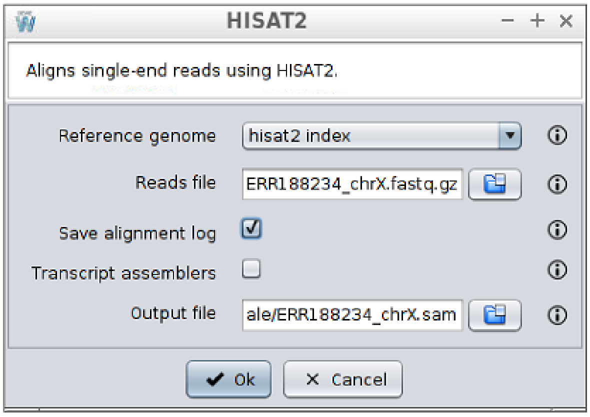
Bowtie2 reads alignment

File: ERR188044_chrX_1.falign.sam

5.3.2.2 HISAT2

Clicking on the *Align > Paired-end Reads > HISAT2* button, a new window will be displayed and the following data will be requested:

- *Reference genome*: the reference genome index to use. It must have been previously built or imported into the application.
- *Reads file*: the reads file. Must be in .fq, .fastq or .fastq.gz format.
- *Save alignment log*: whether the alignment log must be saved or not.
- *Transcript assemblers*: whether to report alignments tailored for transcript assemblers (including Stringtie) or not. With this option, HISAT2 requires longer anchor lengths for de novo discovery of splice sites. This leads to fewer alignments with short-anchors, which helps transcript assemblers improve significantly in computation and memory usage.
- *Output file*: the output file to save the alignments. Should be a .sam file.


HISAT2

Aligns single-end reads using HISAT2.

Reference genome: hisat2 index

Reads file: ERR188234_chrX.fastq.gz

Save alignment log: ☑

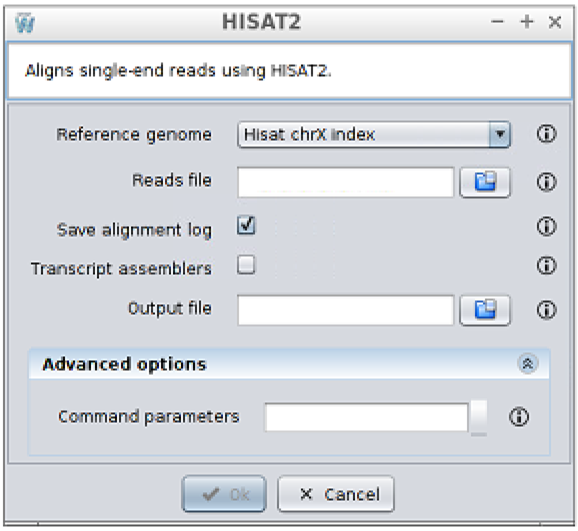
Transcript assemblers: ☐

Output file: ale/ERR188234_chrX.sam

In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is one more parameter:

- *Command parameters*, that allows the user to manually introduce others HISAT2 alignment parameters, like a command line execution. It is important that the parameters are entered correctly or the execution of HISAT2 will fail.

For more information on this advanced options, please, check the HISAT2 reference manual (https://ccb.jhu.edu/software/hisat2/manual.shtml#command-line-1).


HISAT2

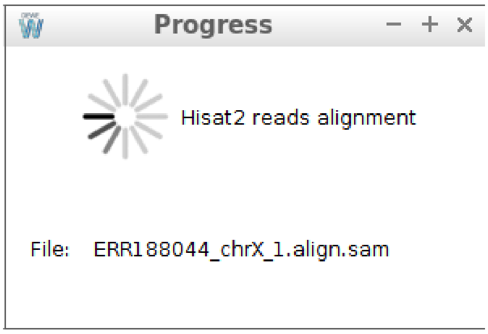
Aligns single-end reads using HISAT2.

Reference genome: Hisat chrX index
Reads file:
Save alignment log: ☑
Transcript assemblers: □
Output file:

Advanced options

Command parameters:

Once the *Ok* button is pressed, a message will be displayed until the end of the process.

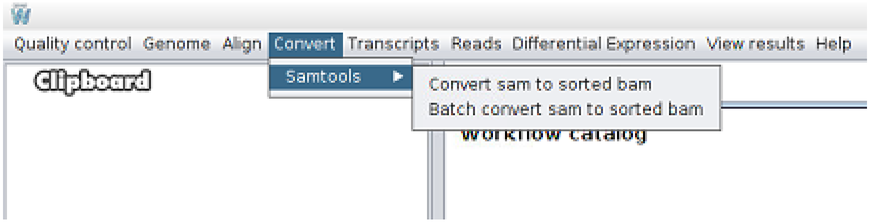

Progress

Hisat2 reads alignment

File: ERR188044_chrX_1.align.sam

5.4 The *Convert* menu

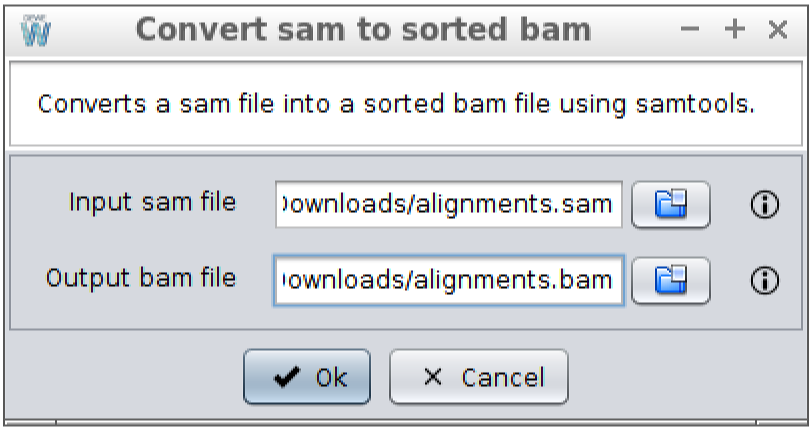
This menu provides operations for converting and sorting read alignments in sam format into bam format using samtools.



5.4.1 Convert sam to sorted bam

Clicking on the *Convert* > *Samtools* > *Convert sam to sorted bam* button, a new window will be displayed and the following data will be requested:

- *Input sam file*: the input sam file. Must be in .sam format.
- *Output bam file*: optionally, an output bam file. If not provided, a file with the same name that the input sam file with ".bam" extension will be used.

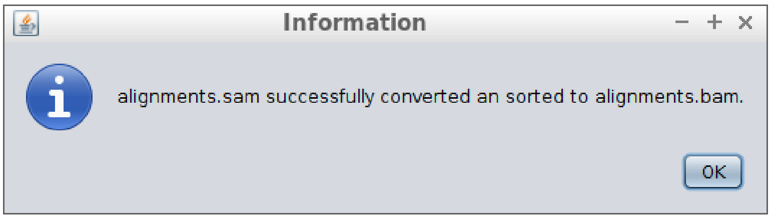


Converts a sam file into a sorted bam file using samtools.

Input sam file:)ownloads/alignments.sam

Output bam file: ownloads/alignments.bam

Once the *Ok* button is pressed, conversion starts and a message will be displayed until the end of the process, when an information message is shown.



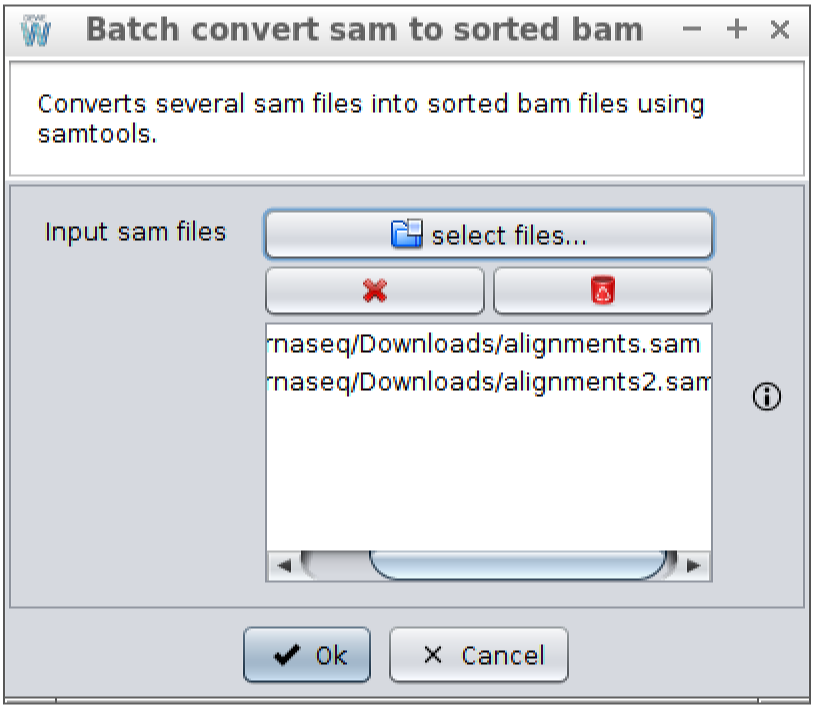
alignments.sam successfully converted an sorted to alignments.bam.

To convert a set of sam files into bam in a single step, the *Batch convert sam to sorted bam* operation is provided.

5.4.2 Batch convert sam to sorted bam

Clicking on the *Convert* > *Samtools* > *Batch convert sam to sorted bam* button, a new window will be displayed and the following data will be requested:

- *Input sam files*: the input sam files. . Must be in .sam format.

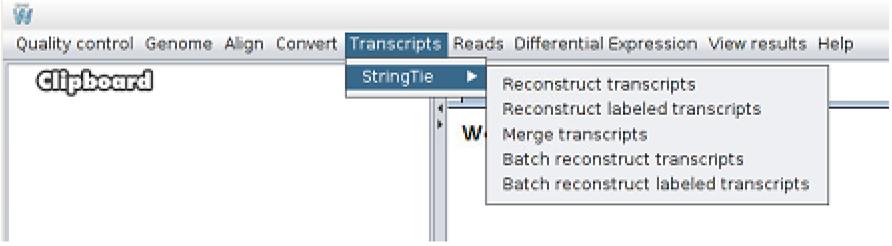


Batch convert sam to sorted bam
Converts several sam files into sorted bam files using samtools.
Input sam files
select files...
rnaseq/Downloads/alignments.sam
rnaseq/Downloads/alignments2.sam
Ok
Cancel

Once the *Ok* button is pressed, conversion of each file starts and one message for each file will be displayed until the end of the process, when an information message is shown.

5.5 The *Transcripts* menu

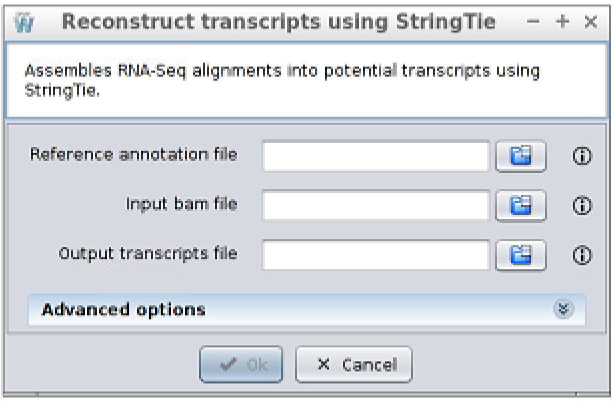
This menu provides operations for obtaining and processing transcripts using StringTie.



5.5.1 Reconstruct transcripts

This operation allows the assembly of RNA-Seq alignments into potential transcripts. Clicking on the *Transcripts* > *StringTie* > *Reconstruct transcripts* button, a new window will be displayed and the following data will be requested:

- *Reference annotation file*: the reference annotation file (.gtf).
- *Input bam file*: the input bam file. . Must be in .bam format.
- *Output transcripts file*: optionally, an output transcripts file (.gtf). If not provided, it will be created in the same directory than the input bam file.



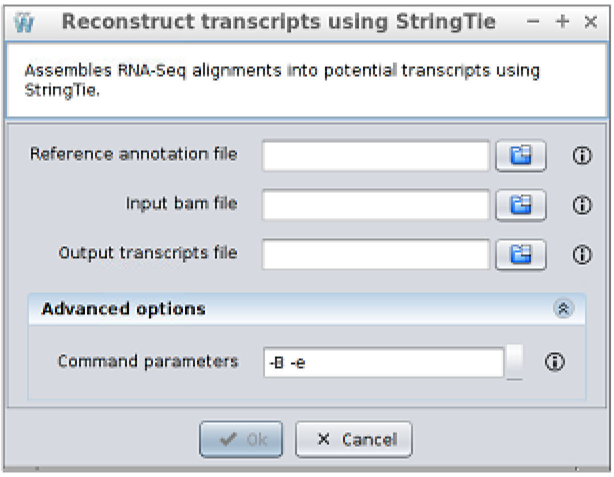
Reconstruct transcripts using StringTie

Assembles RNA-Seq alignments into potential transcripts using StringTie.

In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is one more parameter:

- *Command parameters*, that allows the user to manually introduce others StringTie execution parameters, like a command line execution. It is important that the parameters are entered correctly or the execution of StringTie will fail.

For more information on this advanced options, please, check the HISAT2 reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual).

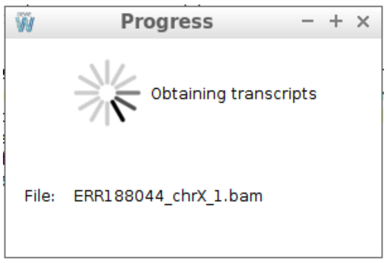


Reconstruct transcripts using StringTie

Assembles RNA-Seq alignments into potential transcripts using StringTie.

Command parameters: -B -e

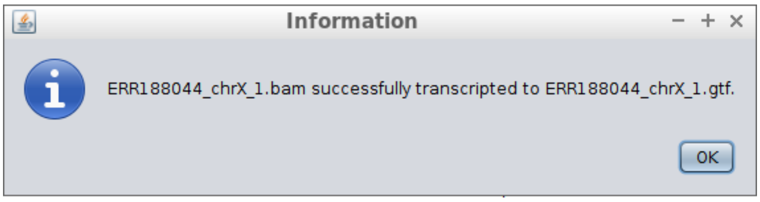
Once the *Ok* button is pressed, StringTie analysis starts and a message will be displayed until the end of the process, when an information message is shown.



Progress

Obtaining transcripts

File: ERR188044_chrX_1.bam



Information

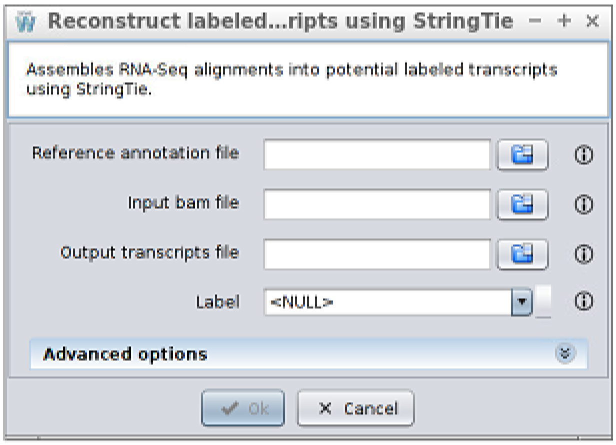
ERR188044_chrX_1.bam successfully transcripted to ERR188044_chrX_1.gtf.

To process a set of bam files in a single step, the *Batch reconstruct transcripts* operation is provided.

5.5.2 Reconstruct labeled transcripts

This operation allows the assembly of RNA-Seq alignments into potential labeled transcripts. Clicking on the *Transcripts* > *StringTie* > *Reconstruct labeled transcripts* button, a new window will be displayed and the following data will be requested:

- *Reference annotation file*: the reference annotation file (.gtf).
- *Input bam file*: the input bam file.. Must be in .bam format.
- *Output transcripts file*: optionally, an output transcripts file (.gtf). If not provided, it will be created in the same directory than the input bam file.
- *Label*: optionally, the label for the -l option of StringTie. This label is the name prefix for output transcripts. If not provided, it will be used the file name.



Reconstruct labeled...ripts using StringTie

Assembles RNA-Seq alignments into potential labeled transcripts using StringTie.

Reference annotation file

Input bam file

Output transcripts file

Label <NULL>

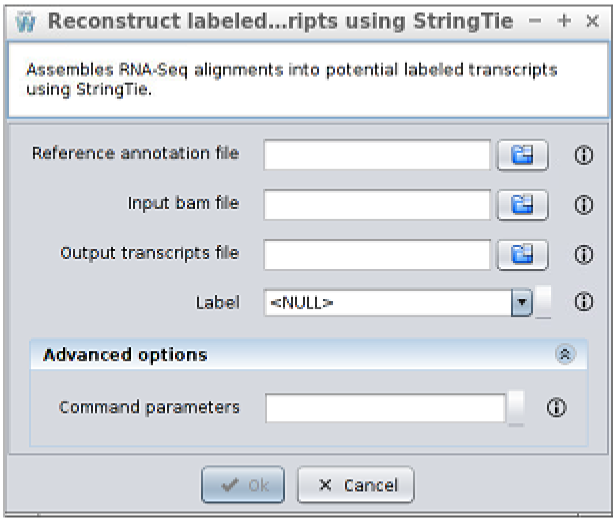
Advanced options

✓ Ok ✕ Cancel

In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is one more parameter:

- *Command parameters*, that allows the user to manually introduce others StringTie execution parameters, like a command line execution. It is important that the parameters are entered correctly or the execution of StringTie will fail.

For more information on this advanced options, please, check the HISAT2 reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual).



Reconstruct labeled...ripts using StringTie

Assembles RNA-Seq alignments into potential labeled transcripts using StringTie.

Reference annotation file

Input bam file

Output transcripts file

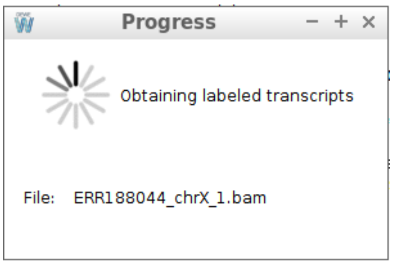
Label <NULL>

Advanced options

Command parameters

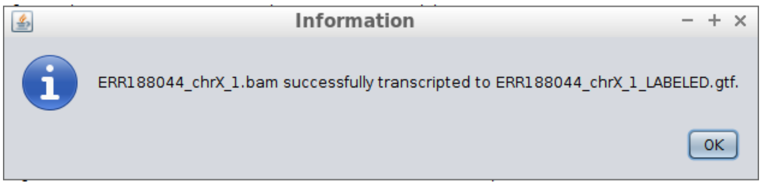
✓ Ok ✕ Cancel

Once the *Ok* button is pressed, StringTie analysis starts and a message will be displayed until the end of the process, when an information message is shown.



Obtaining labeled transcripts

File: ERR188044_chrX_1.bam



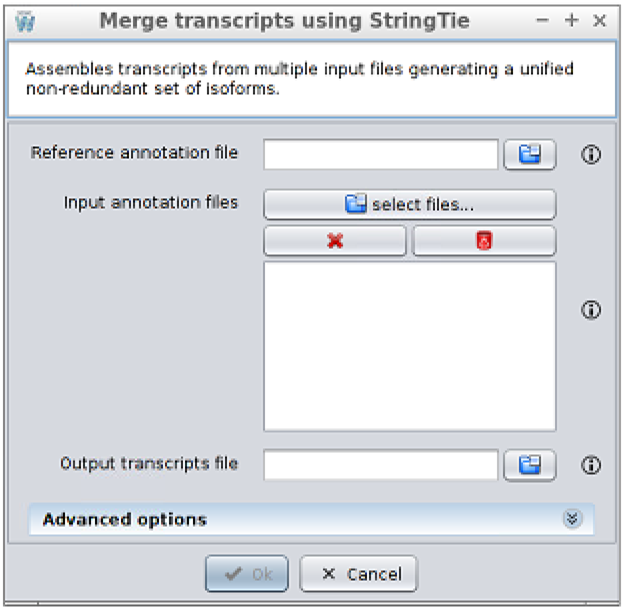
ERR188044_chrX_1.bam successfully transcripted to ERR188044_chrX_1_LABELED.gtf.

To process a set of bam files in a single step, the *Batch reconstruct labeled transcripts* operation is provided.

5.5.3 Merge transcripts

This operation allows the assembly of transcripts from multiple input files to generate a unified non-redundant set of isoforms. Clicking on the *Transcripts > StringTie > Merge transcripts* button, a new window will be displayed and the following data will be requested:

- *Reference annotation file*: the reference annotation file (.gtf).
- *Input annotation file*: the input annotation files (.gtf).
- *Output transcripts file*: Optionally, an output transcripts file (.gtf). If not provided, it will be created in the same directory than the reference annotation file with name *mergedAnnotation.gtf*.

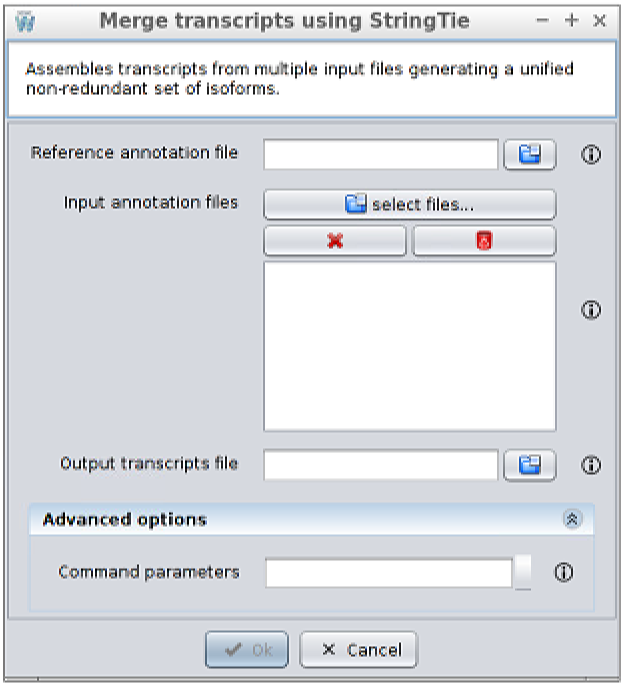


Assembles transcripts from multiple input files generating a unified non-redundant set of isoforms.

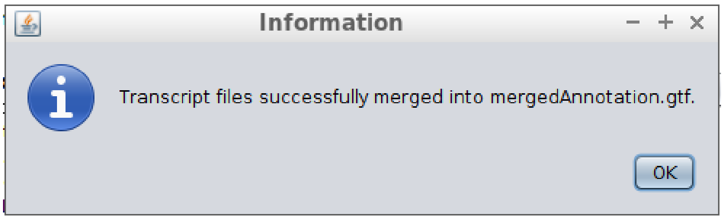
In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is one more parameter:

- *Command parameters*, that allows the user to manually introduce others StringTie execution parameters, like a command line execution. It is important that the parameters are entered correctly or the execution of StringTie will fail.

For more information on this advanced options, please, check the HISAT2 reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual).


Merge transcripts using StringTie
Assembles transcripts from multiple input files generating a unified non-redundant set of isoforms.
Reference annotation file
Input annotation files
select files...
Output transcripts file
Advanced options
Command parameters
Ok
Cancel

Once the *Ok* button is pressed, StringTie analysis starts and a message will be displayed until the end of the process, when an information message is shown.

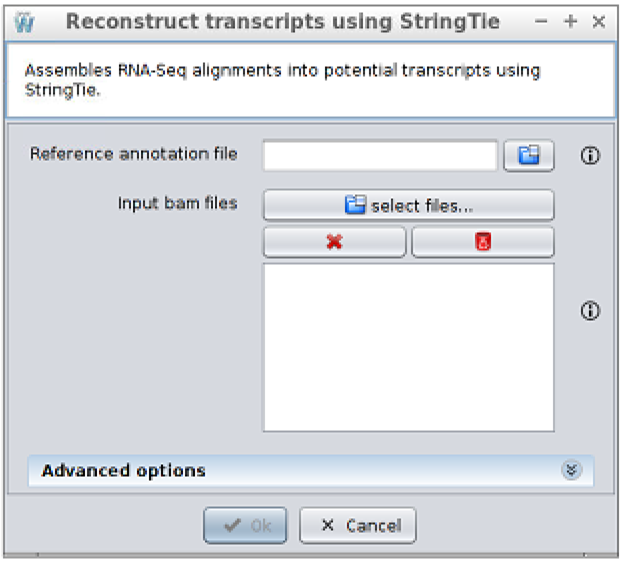


Transcript files successfully merged into mergedAnnotation.gtf.

5.5.4 Batch reconstruct transcripts

This operation allows the assembly of RNA-Seq alignments into potential transcripts. Clicking on the *Transcripts* > *StringTie* > *Batch reconstruct transcripts* button, a new window will be displayed and the following data will be requested:

- *Reference annotation file*: the reference annotation file (.gtf).
- *Input bam file*: the input bam files. Must be in .bam format.



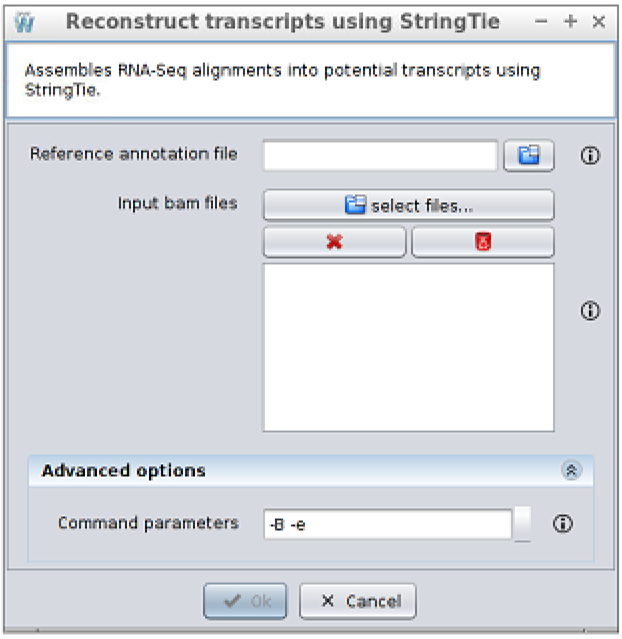
Reconstruct transcripts using StringTie

Assembles RNA-Seq alignments into potential transcripts using StringTie.

In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is one more parameter:

- *Command parameters*, that allows the user to manually introduce others StringTie execution parameters, like a command line execution. It is important that the parameters are entered correctly or the execution of StringTie will fail.

For more information on this advanced options, please, check the HISAT2 reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual).

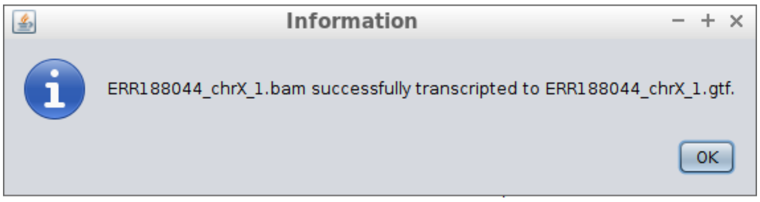


Reconstruct transcripts using StringTie

Assembles RNA-Seq alignments into potential transcripts using StringTie.

Command parameters: -B -e

Once the *Ok* button is pressed, StringTie analysis starts and a message will be displayed until the end of the process, when an information message is shown for each input bam file.



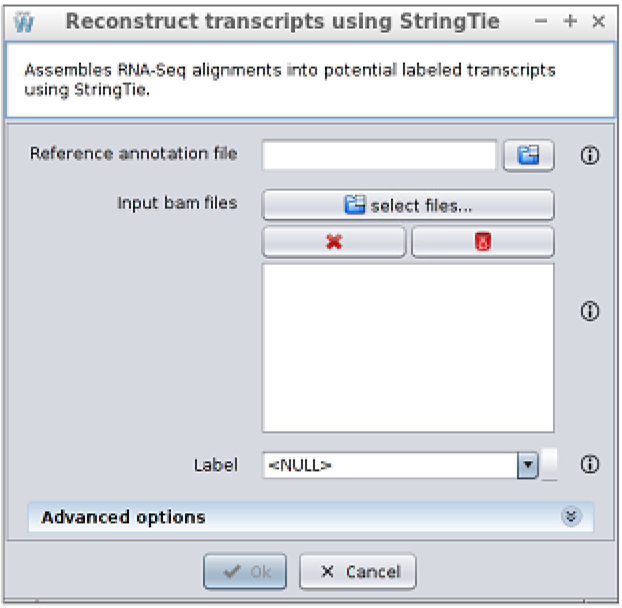
Information

ERR188044_chrX_1.bam successfully transcripted to ERR188044_chrX_1.gtf.

5.5.5 Batch reconstruct labeled transcripts

This operation allows the assembly of RNA-Seq alignments into potential labeled transcripts. Clicking on the *Transcripts* > *StringTie* > *Reconstruct labeled transcripts* button, a new window will be displayed and the following data will be requested:

- *Reference annotation file*: the reference annotation file (.gtf).
- *Input bam files*: the input bam files (.gtf).
- *Label*: optionally, the label for the -l option of StringTie. This label is the name prefix for output transcripts. If not provided, it will be used the file name.



Reconstruct transcripts using StringTie

Assembles RNA-Seq alignments into potential labeled transcripts using StringTie.

Reference annotation file

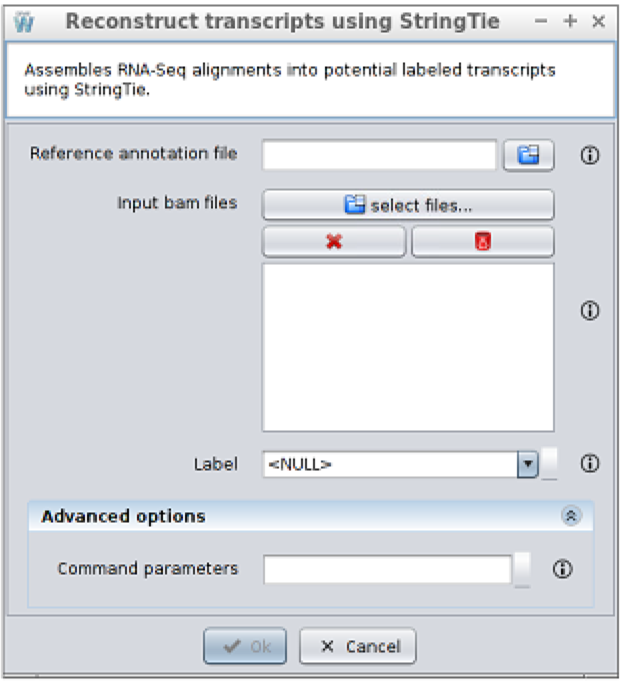
Input bam files

Label <NULL>

In addition to these parameters a menu of *Advanced options* can be displayed. Within this menu there is one more parameter:

- *Command parameters*, that allows the user to manually introduce others StringTie execution parameters, like a command line execution. It is important that the parameters are entered correctly or the execution of StringTie will fail.

For more information on this advanced options, please, check the HISAT2 reference manual (http://ccb.jhu.edu/software/stringtie/index.shtml?t=manual).

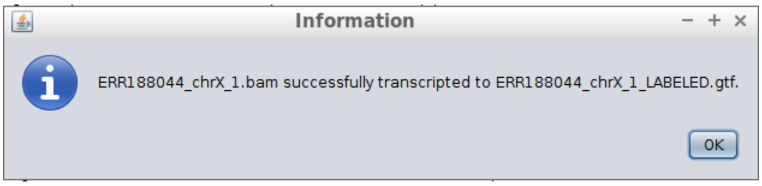


Reconstruct transcripts using StringTie

Assembles RNA-Seq alignments into potential labeled transcripts using StringTie.

Label: <NULL>

Once the *Ok* button is pressed, StringTie analysis starts and a message will be displayed until the end of the process, when an information message is shown for each input bam file.

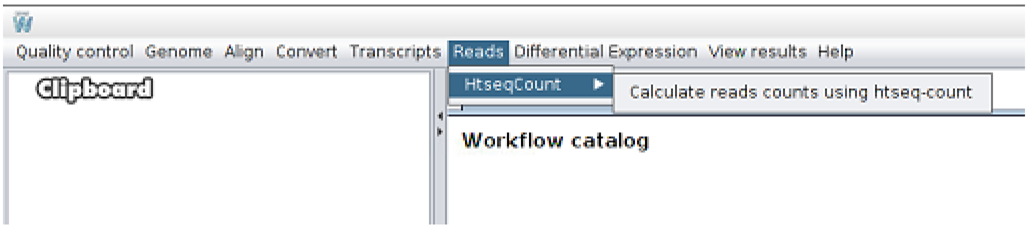


Information

ERR188044_chrX_1.bam successfully transcripted to ERR188044_chrX_1_LABELED.gtf.

5.6 The *Reads* menu

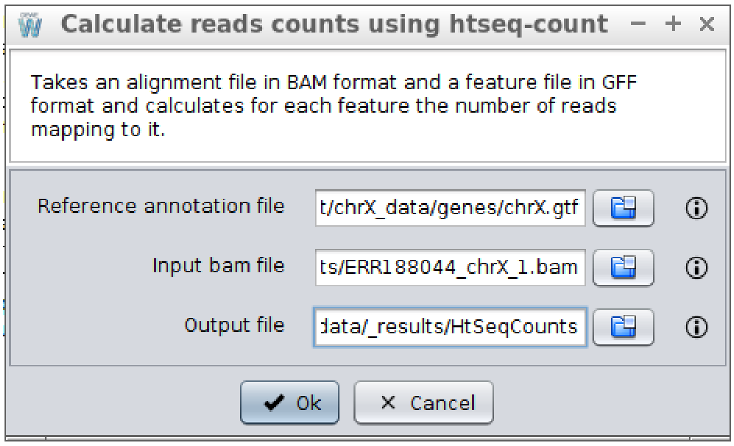
This menu provides other operations dealing with RNA-Seq reads.



5.6.1 Calculate reads counts using htseq-count

This operation takes an alignment file in BAM format and a feature file in GFF format and calculates for each feature the number of reads mapping to it. Clicking on the *Reads* > *HtseqCount* > *Calculate read counts using htseq-count* button, a new window will be displayed and the following data will be requested:

- *Reference annotation file*: the reference annotation file (.gtf).
- *Input bam file*: the input bam file. Must be in .bam format.
- *Output file*: the output file where results are stored.



Calculate reads counts using htseq-count

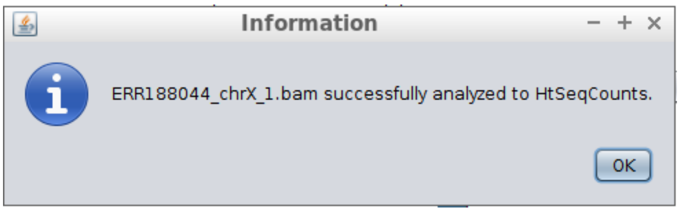
Takes an alignment file in BAM format and a feature file in GFF format and calculates for each feature the number of reads mapping to it.

Reference annotation file: t/chrX_data/genes/chrX.gtf

Input bam file: ts/ERR188044_chrX_1.bam

Output file: data/_results/HtSeqCounts

Once the *Ok* button is pressed, processing starts and a message will be displayed until the end of the process, when an information message is shown.

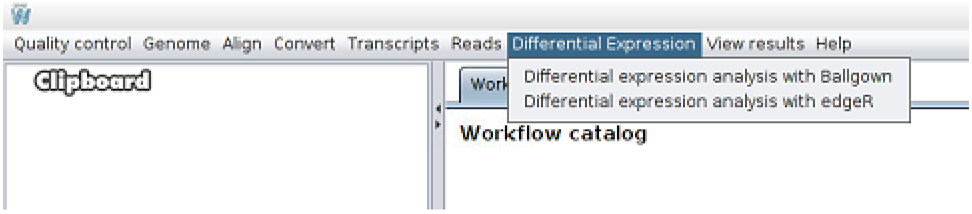


Information

ERR188044_chrX_1.bam successfully analyzed to HtSeqCounts.

5.7 The *Differential Expression* menu

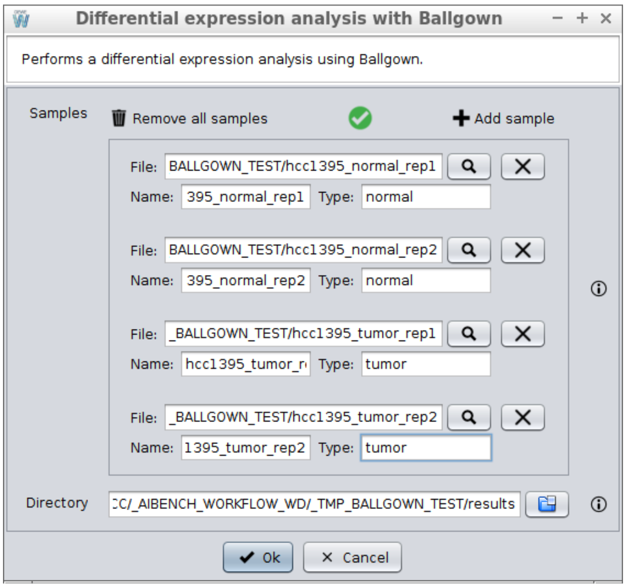
This menu provides operations for performing differential expression analyses.



5.7.1 Ballgown

This operation allows you to perform a differential expression analysis using the Ballgown R library. Clicking on the *Differential Expression* > *Differential expression analysis with Ballgown* button, a new window will be displayed and the following data will be requested:

- *Samples*: the Ballgown samples to analyze. For each sample, you must provide:
    - *File*: the directory where files required by Ballgown are located. The files required by Ballgown are: *e2t.ctab*, *e_data.ctab*, *i2t.ctab*, *i_data.ctab*, and *t_data.ctab*. These files can be produced with StringTie (see subsection *5.4 The Transcripts* menu for more details on using StringTie).
    - *Name*: the name of the sample.
    - *Type*: the type or experimental condition of the sample. Note that this analysis requires two conditions with at least two samples each.
- *Directory*: the output directory where results are stored.



Differential expression analysis with Ballgown

Performs a differential expression analysis using Ballgown.

Samples

File: BALLGOWN_TEST/hcc1395_normal_rep1
Name: 395_normal_rep1 Type: normal

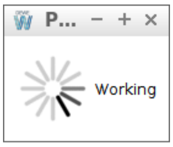
File: BALLGOWN_TEST/hcc1395_normal_rep2
Name: 395_normal_rep2 Type: normal

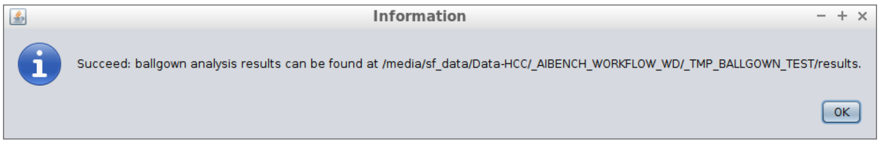
File: _BALLGOWN_TEST/hcc1395_tumor_rep1
Name: hcc1395_tumor_r Type: tumor

File: _BALLGOWN_TEST/hcc1395_tumor_rep2
Name: 1395_tumor_rep2 Type: tumor

Directory: :C/_AIBENCH_WORKFLOW_WD/_TMP_BALLGOWN_TEST/results

Once the *Ok* button is pressed, the analysis starts and a message will be displayed until the end of the process, when an information message is shown. For Ballgown results visualization refer to subsection *6.1 Ballgown*.





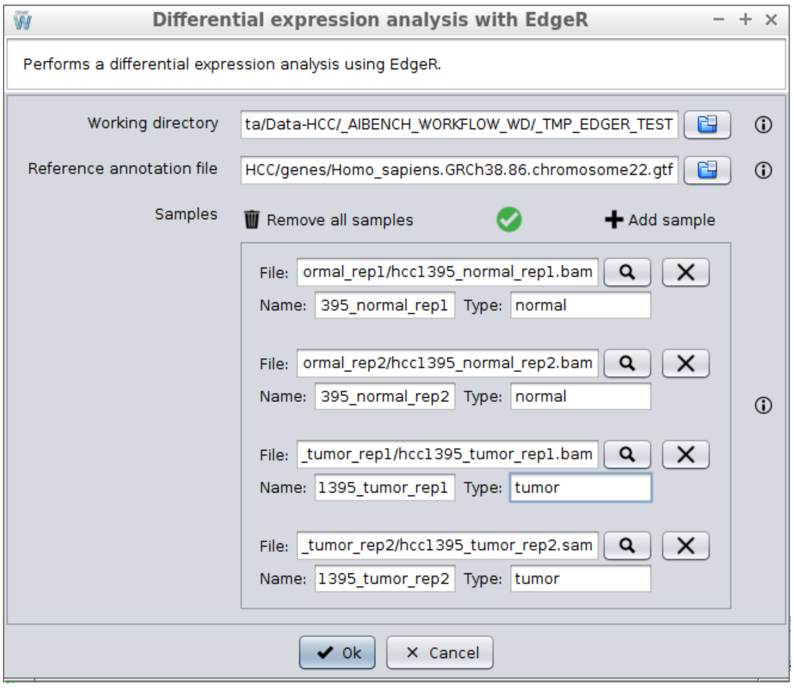
Information

Succeed: ballgown analysis results can be found at /media/sf_data/Data-HCC/_AIBENCH_WORKFLOW_WD/_TMP_BALLGOWN_TEST/results.

5.7.2 edgeR

This operation allows you to perform a differential expression analysis using the edgeR R library. Clicking on the *Differential Expression > Differential expression analysis with edgeR* button, a new window will be displayed and the following data will be requested:

- *Working directory*: the output directory where results are stored.
- *Reference annotation file*: the reference annotation file (.gtf).
- *Samples*: the edgeR samples to analyze. For each sample, you must provide:
  - *File*: the alignment bam file. Must be in .bam format.
  - *Name*: the name of the sample.
  - *Type*: the type or experimental condition of the sample. Note that this analysis requires two conditions with at least two samples each.



Differential expression analysis with EdgeR

Performs a differential expression analysis using EdgeR.

Working directory: ta/Data-HCC/_AIBENCH_WORKFLOW_WD/_TMP_EDGER_TEST

Reference annotation file: HCC/genes/Homo_sapiens.GRCh38.86.chromosome22.gtf

Samples: Remove all samples | Add sample

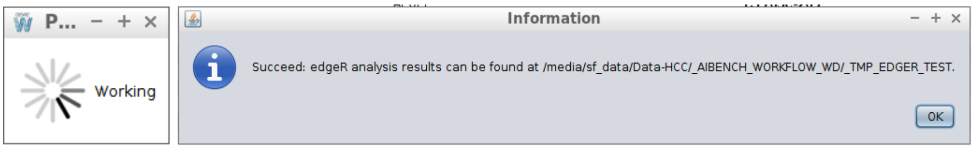
File: ormal_rep1/hcc1395_normal_rep1.bam
Name: 395_normal_rep1 Type: normal

File: ormal_rep2/hcc1395_normal_rep2.bam
Name: 395_normal_rep2 Type: normal

File: _tumor_rep1/hcc1395_tumor_rep1.bam
Name: 1395_tumor_rep1 Type: tumor

File: _tumor_rep2/hcc1395_tumor_rep2.sam
Name: 1395_tumor_rep2 Type: tumor

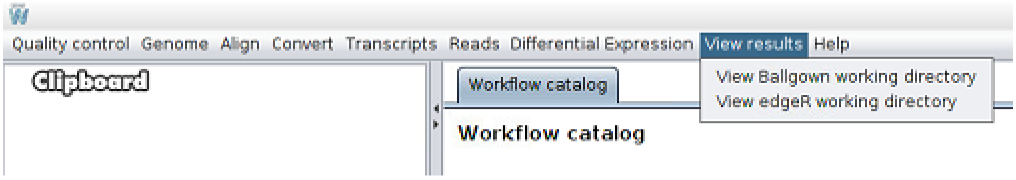
Once the *Ok* button is pressed, the analysis starts and a message will be displayed until the end of the process, when an information message is shown. For edgeR results visualization refer to subsection *6.2 edgeR*.



Succeed: edgeR analysis results can be found at /media/sf_data/Data-HCC/_AIBENCH_WORKFLOW_WD/_TMP_EDGER_TEST.

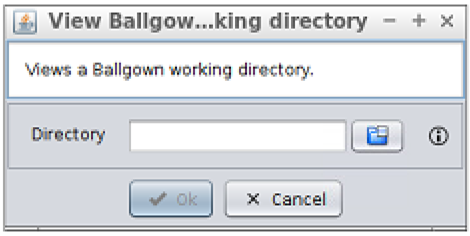
5.8 The *View results* menu

DEWE allows the visualisation of differential expression results previously performed by the tool. Through this menu, the user can visualise the Ballgown (*View Ballgown working directory option*) and edgeR (*View edgeR working directory* option) results contained on a directory.

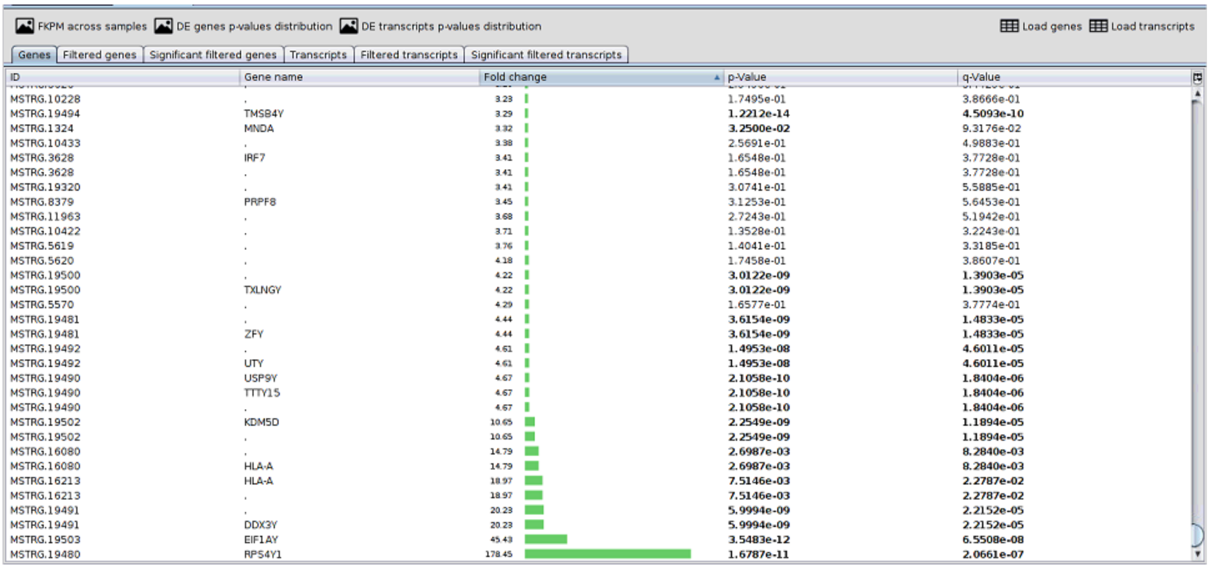


5.8.1 View Ballgown results directory

Clicking on View Ballgown results directory, a new window appear.


View Ballgow...king directory
Views a Ballgown working directory.
Directory
Ok
Cancel

In this new window, the user must select the directory where the results of Ballgown are stored and click on the OK button. Once this is done, the six analysis performed by Ballgown will be displayed.

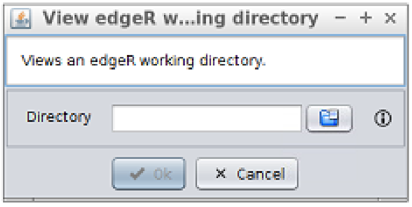

FKPM across samples
DE genes p-values distribution
DE transcripts p-values distribution
Load genes
Load transcripts
Genes
Filtered genes
Significant filtered genes
Transcripts
Filtered transcripts
Significant filtered transcripts

ID	Gene name	Fold change	p-Value	q-Value
MSTRG.10228	.	3.23	1.7495e-01	3.8666e-01
MSTRG.19494	TMSB4Y	3.29	1.2212e-14	4.5093e-10
MSTRG.1324	MNDA	3.32	3.2500e-02	9.3176e-02
MSTRG.10433	.	3.38	2.5691e-01	4.9883e-01
MSTRG.3628	IRF7	3.41	1.6548e-01	3.7728e-01
MSTRG.3628	.	3.41	1.6548e-01	3.7728e-01
MSTRG.19320	.	3.41	3.0741e-01	5.5885e-01
MSTRG.8379	PRPF8	3.45	3.1253e-01	5.6453e-01
MSTRG.11963	.	3.68	2.7243e-01	5.1942e-01
MSTRG.10422	.	3.71	1.3528e-01	3.2243e-01
MSTRG.5619	.	3.76	1.4041e-01	3.3185e-01
MSTRG.5620	.	4.18	1.7458e-01	3.8607e-01
MSTRG.19500	.	4.22	3.0122e-09	1.3903e-05
MSTRG.19500	TXLNGY	4.22	3.0122e-09	1.3903e-05
MSTRG.5570	.	4.29	1.6577e-01	3.7774e-01
MSTRG.19481	.	4.44	3.6154e-09	1.4833e-05
MSTRG.19481	ZFY	4.44	3.6154e-09	1.4833e-05
MSTRG.19492	.	4.61	1.4953e-08	4.6011e-05
MSTRG.19492	UTY	4.61	1.4953e-08	4.6011e-05
MSTRG.19490	USP9Y	4.67	2.1058e-10	1.8404e-06
MSTRG.19490	TTTY15	4.67	2.1058e-10	1.8404e-06
MSTRG.19490	.	4.67	2.1058e-10	1.8404e-06
MSTRG.19502	KDM5D	10.65	2.2549e-09	1.1894e-05
MSTRG.19502	.	10.65	2.2549e-09	1.1894e-05
MSTRG.16080	.	14.79	2.6987e-03	8.2840e-03
MSTRG.16080	HLA-A	14.79	2.6987e-03	8.2840e-03
MSTRG.16213	HLA-A	18.97	7.5146e-03	2.2787e-02
MSTRG.16213	.	18.97	7.5146e-03	2.2787e-02
MSTRG.19491	.	20.23	5.9994e-09	2.2152e-05
MSTRG.19491	DDX3Y	20.23	5.9994e-09	2.2152e-05
MSTRG.19503	EIF1AY	45.43	3.5483e-12	6.5508e-08
MSTRG.19480	RPS4Y1	178.45	1.6787e-11	2.0661e-07

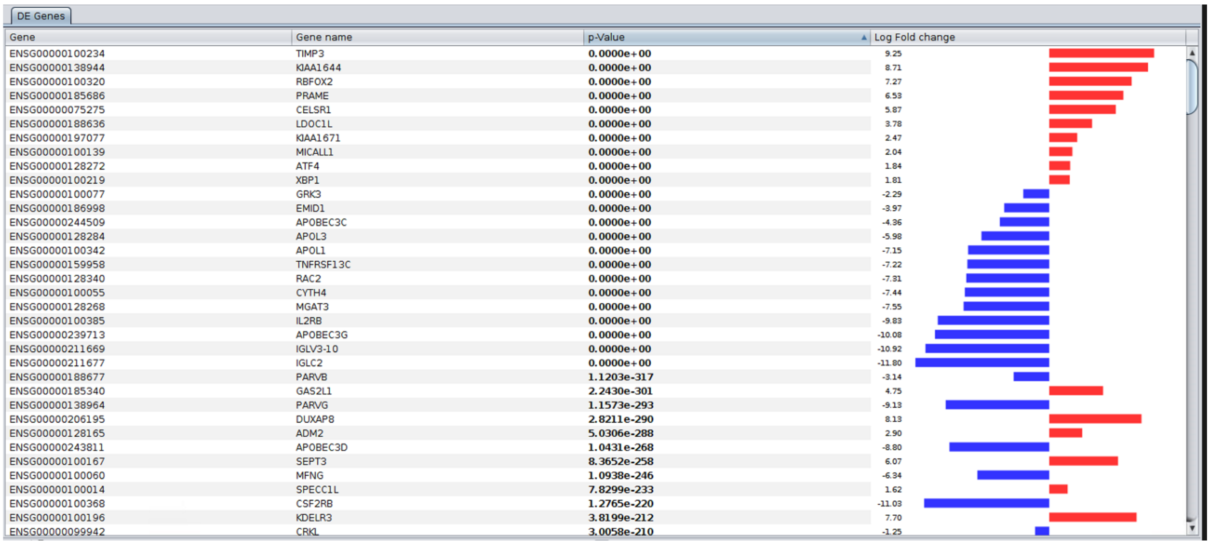
The interaction with these results is explained in section 6.1.2

5.8.2 View edgeR results directory

Clicking on *View edgeR results directory*, a new window appear.


View edgeR w...ing directory
Views an edgeR working directory.
Directory
Ok
Cancel

In this new window, the user must select the directory where the results of edgeR are stored and click on the OK button. Once this is done, the analysis performed by edgeR will be displayed.



Gene	Gene name	p-Value	Log Fold change
ENSG00000100234	TIMP3	0.0000e+00	9.25
ENSG00000138944	KIAA1644	0.0000e+00	8.71
ENSG00000100320	RBFOX2	0.0000e+00	7.27
ENSG00000185686	PRAME	0.0000e+00	6.53
ENSG00000075275	CELSR1	0.0000e+00	5.87
ENSG00000188636	LDOC1L	0.0000e+00	3.78
ENSG00000197077	KIAA1671	0.0000e+00	2.47
ENSG00000100139	MICALL1	0.0000e+00	2.04
ENSG00000128272	ATF4	0.0000e+00	1.84
ENSG00000100219	XBP1	0.0000e+00	1.81
ENSG00000100077	GRK3	0.0000e+00	-2.29
ENSG00000186998	EMID1	0.0000e+00	-3.97
ENSG00000244509	APOBEC3C	0.0000e+00	-4.36
ENSG00000128284	APOL3	0.0000e+00	-5.98
ENSG00000100342	APOL1	0.0000e+00	-7.15
ENSG00000159958	TNFRSF13C	0.0000e+00	-7.22
ENSG00000128340	RAC2	0.0000e+00	-7.31
ENSG00000100055	CYTH4	0.0000e+00	-7.44
ENSG00000128268	MGAT3	0.0000e+00	-7.55
ENSG00000100385	IL2RB	0.0000e+00	-9.83
ENSG00000239713	APOBEC3G	0.0000e+00	-10.08
ENSG00000211669	IGLV3-10	0.0000e+00	-10.92
ENSG00000211677	IGLC2	0.0000e+00	-11.80
ENSG00000188677	PARVB	1.1203e-317	-3.14
ENSG00000185340	GAS2L1	2.2430e-301	4.75
ENSG00000138964	PARVG	1.1573e-293	-9.13
ENSG00000206195	DUXAP8	2.8211e-290	8.13
ENSG00000128165	ADM2	5.0306e-288	2.90
ENSG00000243811	APOBEC3D	1.0431e-268	-8.80
ENSG00000100167	SEPT3	8.3652e-258	6.07
ENSG00000100060	MFNG	1.0938e-246	-6.34
ENSG00000100014	SPECC1L	7.8299e-233	1.62
ENSG00000100368	CSF2RB	1.2765e-220	-11.03
ENSG00000100196	KDELR3	3.8199e-212	7.70
ENSG00000099942	CRKL	3.0058e-210	-1.25

The interaction with these results is explained in section 6.2.2.

6. Outputs and visualisation

6.1 Ballgown

The Ballgown differential expression analysis can be run from the menu operation (section 5.7.1) or as part of the available workflows (sections 4.2 and 4.3). The outputs are similar in In both cases. This section explains these outputs and the visualisation capabilities of the tool, with a brief indication on the interpretation and the biological relevance of the results shown in the figures.

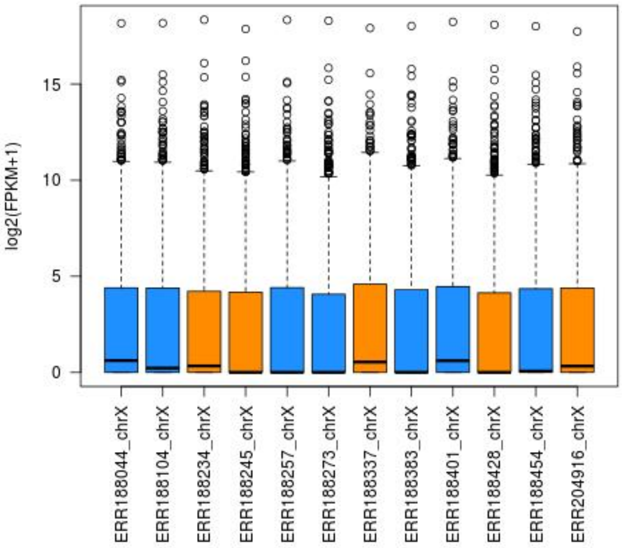
6.1.1 Ballgown outputs

Ballgown will produce several archives that can be useful for the interpretation of the results of the differential expression experiment. The following files are generated:

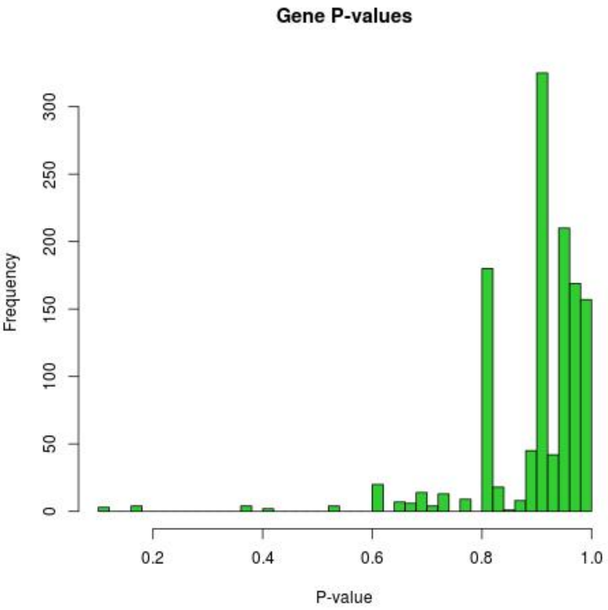
- *consensuspathdb_enrichment_analysis.csv*: this file will list the genes over/underexpressed with their corresponding fold-changes in log2 values. This file, or a subset of it, can be used as input file for ConsensusPathDB gene set enrichment analysis (http://cpdb.molgen.mpg.de/) or DAVID (https://david.ncifcrf.gov/). Using this file and those services, the user can group the genes over or underexpressed in, genetic, metabolic, signaling, gene regulatory or biochemical pathways among others. The user can also perform functional annotation and gene functional classification of the set of genes changing their expression values under given experimental settings.
- *phenotype-data_gene_results.tsv*: a list of the differentially and non-differentially expressed genes between the two conditions set by the user.
- *phenotype-data_gene_results_filtered.tsv*: a subset of the *phenotype-data_gene_results.tsv* file in which only the differentially expressed genes between the two conditions (p-value < 0.05) are listed. In this file, low-abundance genes and genes with variance values across the samples lower than 1 are also filtered and not included in this file.
- *phenotype-data_gene_results_sig.tsv*: a subset of the *phenotype-data_gene_results_filtered.tsv*: file in which only differentially expressed genes between the two conditions, with q-values lower than 0.05 are listed. Usually, this file is used to set a threshold in the fold-change values and define, from a biological point of view, which genes are considered over or under-expressed. Generally, fold-change values (which are given in log2 changes) higher than 1 or lower than one, which correspond to fold-changes higher/lower than 2, are considered. In this point, the user should evaluate whether genes with small fold-change values but consistent q-values, could have a great biological effect (i.e. transcription regulators)
- *phenotype-data_transcript_results.tsv*: a list of the differentially and non-differentially expressed transcripts between the two conditions set by the user. User should consider that transcripts comprise annotated genes and other regions of the genome that produces other molecules such as microRNAs or completely unknown RNAs.

For instance, the last human genome assembly (hg19) comprises roughly 21000 genes but more than 70000 transcripts. A list of the eukaryotic genomes compatible with DEWE can be retrieved in the following link: https://genome.ucsc.edu/FAQ/FAQreleases.html

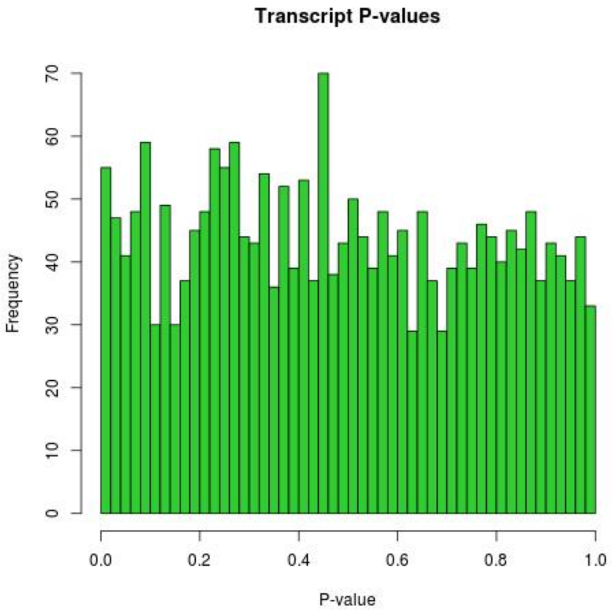
- *phenotype-data_transcript_results_filtered.tsv*: a subset of the *phenotype-data_transcript_results.tsv* file in which only the differentially expressed transcripts between the two conditions (p-value < 0.05) are listed. In this file, low-abundance genes and genes with variance values across the samples lower than 1 are also filtered and not included in this file.
- *phenotype-data_transcript_results_sig.tsv*: a subset of the *phenotype-data_transcript_results_filtered.tsv*: file in which only differentially expressed transcripts between the two conditions, with q-values lower than 0.05 are listed. Usually, this file is used to set a threshold in the fold-change values and define, from a biological point of view, which genes are considered over or under-expressed.
- *FPKM-distribution-across-samples.jpeg*: plot of the distribution of FPKM values across the samples. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder. This figure will provide the user with an indication of the FPKM value distribution in the different samples analysed. As a guideline, FPKM represent a transformation of the raw differential expression values and are expected to be log-normally distributed (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1240081/).



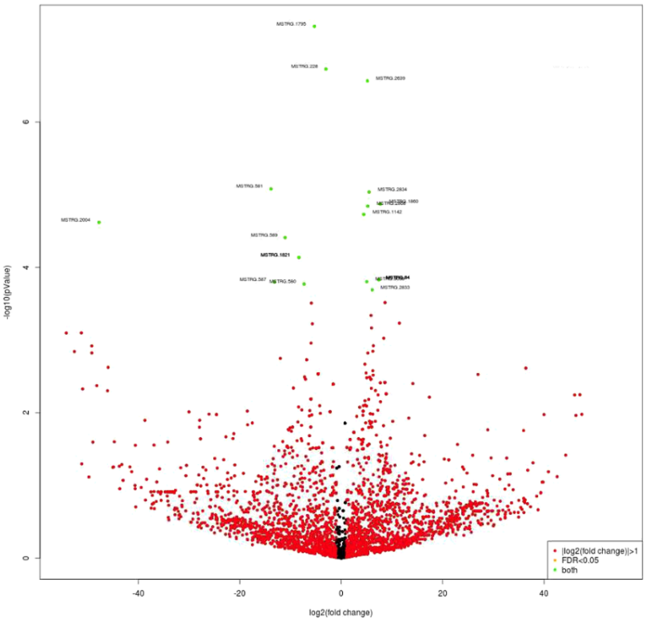
- *genes-DE-pValues-distribution.jpeg*: histogram of the frequencies of experimental p-values corresponding to the genes contained in the analysed genome. It will provide the user with a distribution of the differential expression p-values for all the genes, and it will also provide an indication of the subset of genes without significative differences in their fold-changes (p-value > 0.05). At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder.



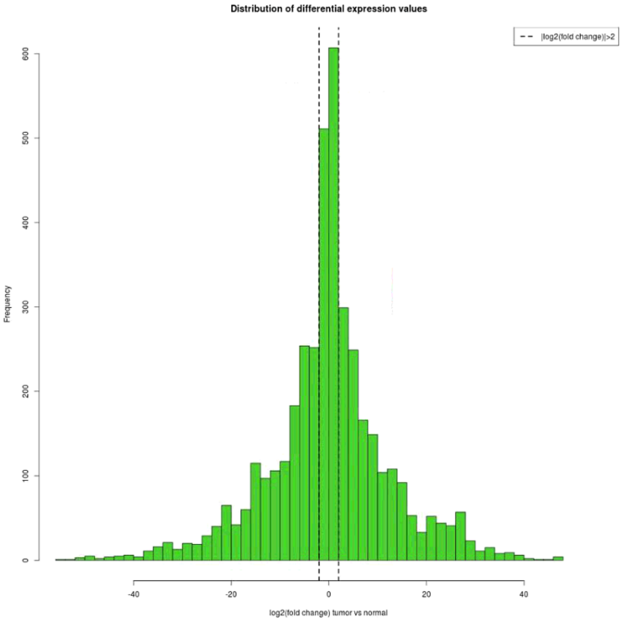
- *transcripts-DE-pValues-distribution.jpeg*: histogram of the frequencies of experimental p-values corresponding to the transcripts contained in the analysed genome. It will provide the user with a distribution of the differential expression p-values for all the transcripts, and it will also provide an indication of the subset of genes without significative differences in their fold-changes (p-value > 0.05). At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder.



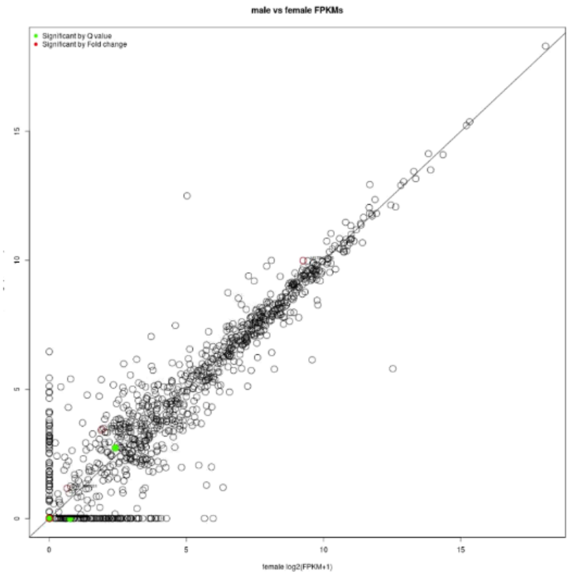
- *volcano-plot.jpeg*: This graphic combines the results of the p-values obtained after statistical tests with every single fold change. It will allow a rapid identification of the genes that increases both their magnitude of fold change and their statistical significance. Orange color represent up- or down-regulated genes (absolute value of log2fold-change > 1). Red color represent genes changing their expression with a statistical significance equal or lower than 0.05, measured as the adjusted p-value using the false discovery rate method. Green color represent up- or down-regulated genes with combine the two previous criteria. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder.



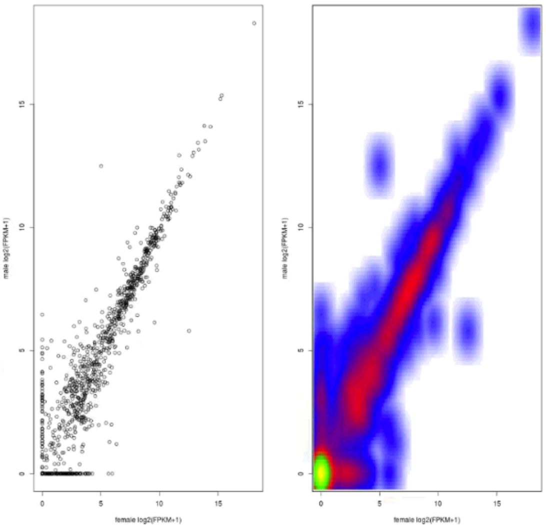
- *DE-values-distribution.jpeg*: distribution of differential expression values histogram, which shows the frequency of the different expression fold-changes in a given experiment. This will give the user an idea of the proportion of genes changing their expression over a given fold-change threshold, which is depicted with a dashed line. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder.



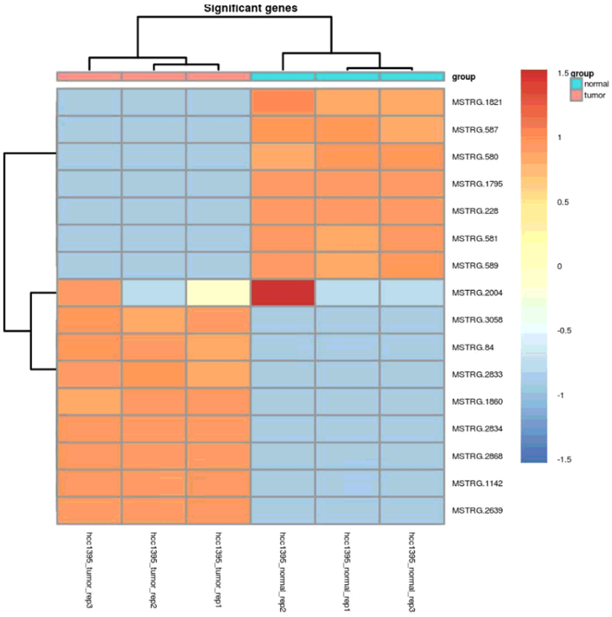
- *FPKM-conditions-correlation.jpeg:* FPKMs correlation between conditions, which are plotted in the x/y axis and will highlight those genes with enriched FPKM values in one or other experimental condition. Genes showing no variation in their experimental FPKM values will adjust to the diagonal line. Note that values are expressed in FPKM+1 values, avoiding dividing by zero when calculating fold changes. Red color represent up- or down-regulated genes (absolute value of log2fold-change > 2). Red color represent genes changing their expression with a statistical significance equal or lower than 0.05, measured as the adjusted p-value using the false discovery rate method. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder.



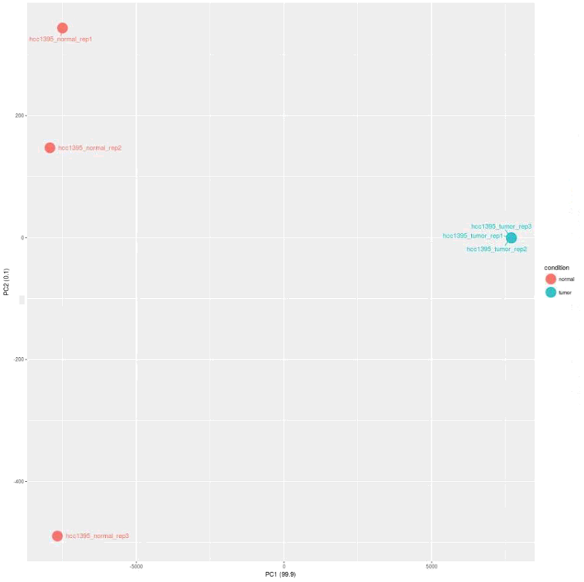
- *FPKM-conditions-density.jpeg*: FPKM correlation between conditions, represented as a density plot. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder.



- *Heatmap.jpeg:* Heatmap of statistically significant genes (q-value<0.05), which a computed row and column clustering, the latter with an additional layer including the phenotype data. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder.

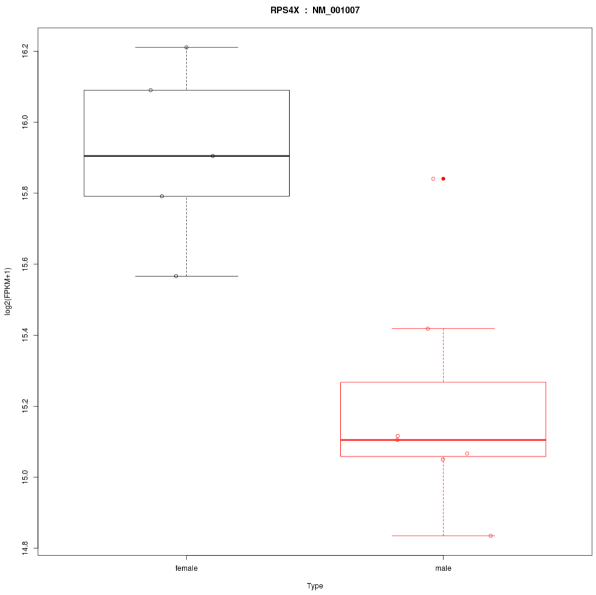


- *pca.jpeg:* Principal Component Analysis plot, in which the global variance of the experiment is decomposed and grouped into new and orthogonal variables denominated components. PCA allows a quick inspection on the factors driving the observed variance, including unsupervised discovery of confounding factors. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the Ballgown results folder.

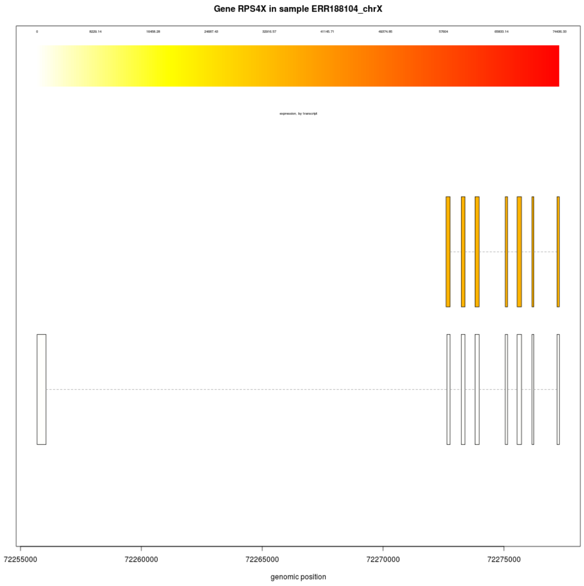


In addition, the following analysis can be performed using the *Ballgown results viewer*, as the next subsection describes:

- *user-images/FPKM-distribution-gene_genename-transcript_transcriptname.format*: FPKM distributions for a given transcript, representing how the over or underexpression values are distributed between the two experimental conditions. This box/jitter plot will give the user an idea of the variation of the differential expression values across the quartiles and around the median. This image can be exported in jpeg, tiff or png and in the resolution and color (colored or grayscale) defined by the user.



- *user-images/transcripts-gene_genename-sample_samplename.format*: Structure and expression levels of the different isoform for a given gene in a given sample. Basically, the user will identify where RNA-Seq reads map with the different gene exons. Eventually, the user might identify potential and differential isoform production between the conditions. This image can be exported in jpeg, tiff or png and in the resolution and color (colored or grayscale) defined by the user.



- *user-tables/transcript_results_sig_percentage.tsv:* a list of differentially expressed transcripts between the two experimental conditions, in which a threshold p-value is empirically set by the user.

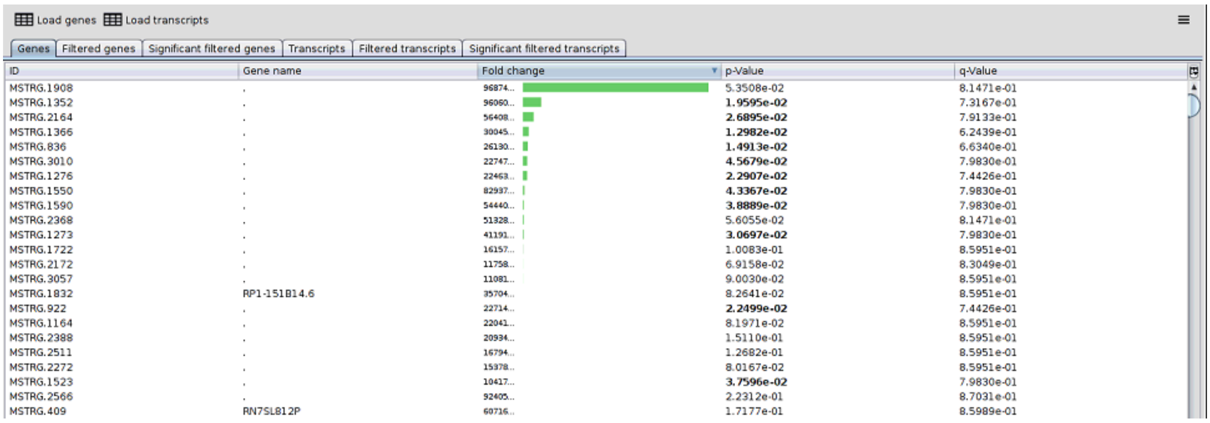
- *user-tables/gene_results_sig_percentage.tsv:* a list of differentially expressed genes between the two experimental conditions, in which a threshold p-value is empirically set by the user.
- A list, named by the user, with the selected number of the most overexpressed or underexpressed genes. This list can serve as input for ConsensusPathDB over-representation gene set analysis or other functional annotation web services such as DAVID, which has been described under the "*consensuspathdb_enrichment_analysis.csv*" section.

6.1.2 Results visualisation

The viewer enables the interactive browsing of genes and transcripts analysis through some generated tables to better understand and visualisation this analysis in the Ballgown working directory.

As you be seen in the following image, this view contains the following six tabs:

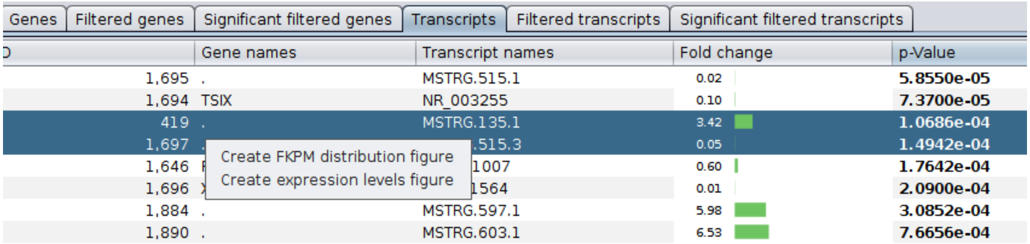
- *Genes*: this tab contains a table with the genes in file *phenotype-data_gene_results.tsv*.
- *Filtered genes*: this tab contains a table with the genes in file *phenotype-data_gene_results_filtered.tsv*.
- *Significant filtered genes*: this tab contains a table with the genes in file *phenotype-data_gene_results_sig.tsv*.
- *Transcripts*: this tab contains a table with the transcripts in file *phenotype-data_transcript_results.tsv*.
- *Filtered transcripts*: this tab contains a table with the transcripts in file *phenotype-data_transcript_results_filtered.tsv*.
- *Significant filtered transcripts*: this tab contains a table with the transcripts in file *phenotype-data_transcript_results_sig.tsv*.



ID	Gene name	Fold change	p-Value	q-Value
MSTRG.1908	.	96874...	5.3508e-02	8.1471e-01
MSTRG.1352	.	96060...	**1.9595e-02**	7.3167e-01
MSTRG.2164	.	56408...	**2.6895e-02**	7.9133e-01
MSTRG.1366	.	30045...	**1.2982e-02**	6.2439e-01
MSTRG.836	.	26130...	**1.4913e-02**	6.6340e-01
MSTRG.3010	.	22747...	**4.5679e-02**	7.9830e-01
MSTRG.1276	.	22463...	**2.2907e-02**	7.4426e-01
MSTRG.1550	.	82937...	**4.3367e-02**	7.9830e-01
MSTRG.1590	.	54440...	**3.8889e-02**	7.9830e-01
MSTRG.2368	.	51328...	5.6055e-02	8.1471e-01
MSTRG.1273	.	41191...	**3.0697e-02**	7.9830e-01
MSTRG.1722	.	16157...	1.0083e-01	8.5951e-01
MSTRG.2172	.	11758...	6.9158e-02	8.3049e-01
MSTRG.3057	.	11081...	9.0030e-02	8.5951e-01
MSTRG.1832	RP1-151B14.6	35704...	8.2641e-02	8.5951e-01
MSTRG.922	.	22714...	**2.2499e-02**	7.4426e-01
MSTRG.1164	.	22041...	8.1971e-02	8.5951e-01
MSTRG.2388	.	20934...	1.5110e-01	8.5951e-01
MSTRG.2511	.	16794...	1.2682e-01	8.5951e-01
MSTRG.2272	.	15378...	8.0167e-02	8.5951e-01
MSTRG.1523	.	10417...	**3.7596e-02**	7.9830e-01
MSTRG.2566	.	92405...	2.2312e-01	8.7031e-01
MSTRG.409	RN7SL812P	60716...	1.7177e-01	8.5989e-01

6.1.2.1 Creation of additional results from transcripts tables

Any of the transcripts tables allows to generate the first two additional analysis described in section *6.1.1 Ballgown outputs*. To perform this analysis, first the rows corresponding to the transcripts that are wanted to export must be selected. Then, right-click must be done in order to make visible a contextual menu with options to create the figures that explain these analysis.

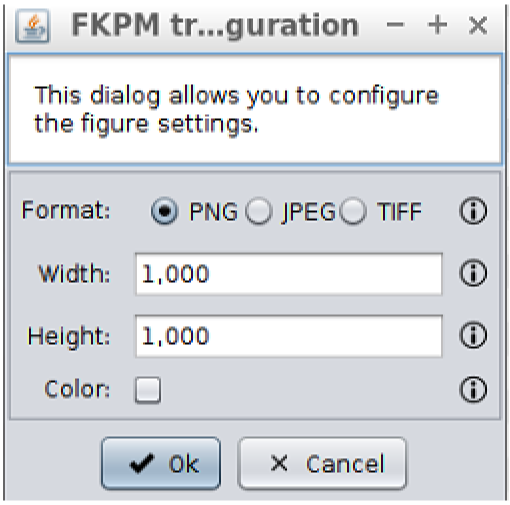


Genes | Filtered genes | Significant filtered genes | Transcripts | Filtered transcripts | Significant filtered transcripts

D	Gene names	Transcript names	Fold change	p-Value
1,695	.	MSTRG.515.1	0.02	5.8550e-05
1,694	TSIX	NR_003255	0.10	7.3700e-05
419	.	MSTRG.135.1	3.42 🟩	1.0686e-04
1,697	.	.515.3	0.05	1.4942e-04
1,646	F	1007	0.60 🟩	1.7642e-04
1,696	X	1564	0.01	2.0900e-04
1,884	.	MSTRG.597.1	5.98 🟩	3.0852e-04
1,890	.	MSTRG.603.1	6.53 🟩	7.6656e-04

Create FKPM distribution figure
Create expression levels figure

Clicking the *Create FKPM distribution figure* the following dialog will appear, allowing to select the image format, resolution and color. After clicking the *Ok* button, the images will be generated inside a directory called *user-images* placed in the working directory.



FKPM tr...guration

This dialog allows you to configure the figure settings.

Format: ● PNG ○ JPEG ○ TIFF

Width: 1,000

Height: 1,000

Color: □

Clicking the *Create expression levels figure* the following dialog will appear, allowing to select the image format, resolution and color along with the samples for which the figure should be generated. After clicking the *Ok* button, the images will be generated inside a directory called *user-images* placed in the working directory.

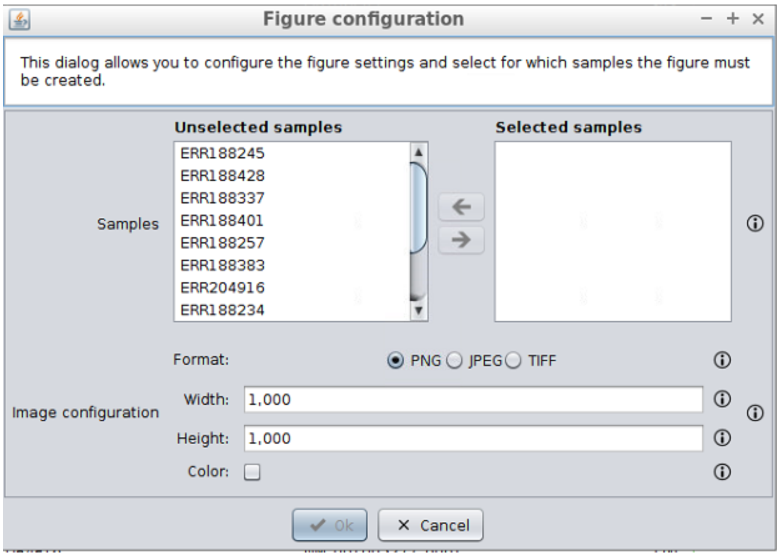


Figure configuration

This dialog allows you to configure the figure settings and select for which samples the figure must be created.

Samples

Unselected samples: ERR188245, ERR188428, ERR188337, ERR188401, ERR188257, ERR188383, ERR204916, ERR188234

Selected samples:

Image configuration

Format: ● PNG ○ JPEG ○ TIFF

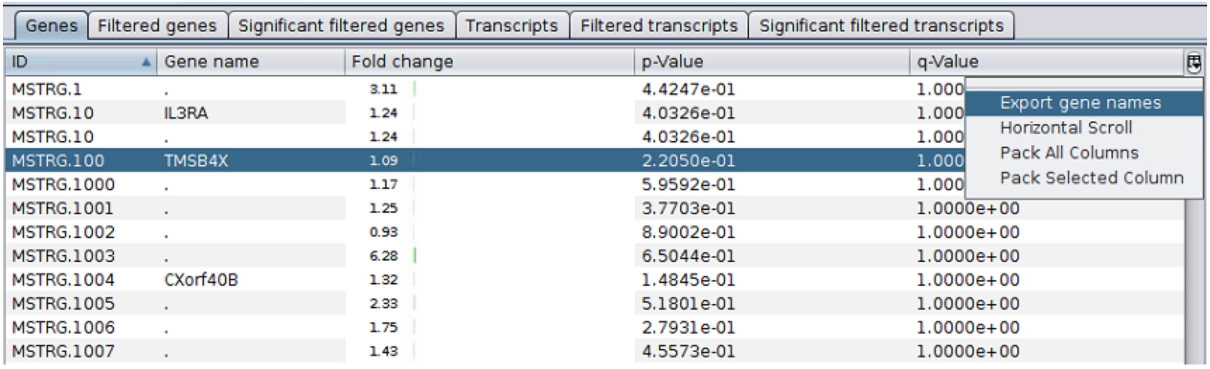
Width: 1,000

Height: 1,000

Color: □

6.1.2.2 Creation of additional results from genes tables

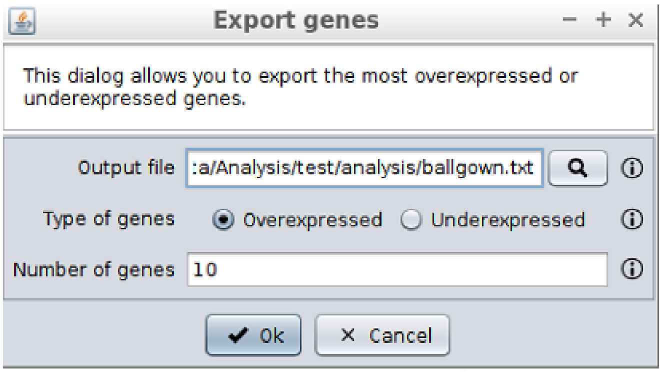
Any of the gene tables allows to generate the last additional analysis described in section *6.1.1 Ballgown outputs*. To perform this analysis, first the button over the vertical scroll must be clicked. Then, the *Export gene names* option must be selected. This will open a new window.



ID	Gene name	Fold change	p-Value	q-Value
MSTRG.1	.	3.11	4.4247e-01	1.000
MSTRG.10	IL3RA	1.24	4.0326e-01	1.000
MSTRG.10	.	1.24	4.0326e-01	1.000
MSTRG.100	TMSB4X	1.09	2.2050e-01	1.000
MSTRG.1000	.	1.17	5.9592e-01	1.000
MSTRG.1001	.	1.25	3.7703e-01	1.0000e+00
MSTRG.1002	.	0.93	8.9002e-01	1.0000e+00
MSTRG.1003	.	6.28	6.5044e-01	1.0000e+00
MSTRG.1004	CXorf40B	1.32	1.4845e-01	1.0000e+00
MSTRG.1005	.	2.33	5.1801e-01	1.0000e+00
MSTRG.1006	.	1.75	2.7931e-01	1.0000e+00
MSTRG.1007	.	1.43	4.5573e-01	1.0000e+00

On this new window the following data will be requested: The first is the Output file, that is the path in which the result file of the analysis will be generated (if a name is not specified for the file, DEWE will name it *ballgown.txt*). The second field is the type of gene expression, overexpressed or underexpressed. And finally, the number of over/underexpressed genes.

After clicking the *Ok* button, the analysis will be generated in the selected path.



This dialog allows you to export the most overexpressed or underexpressed genes.

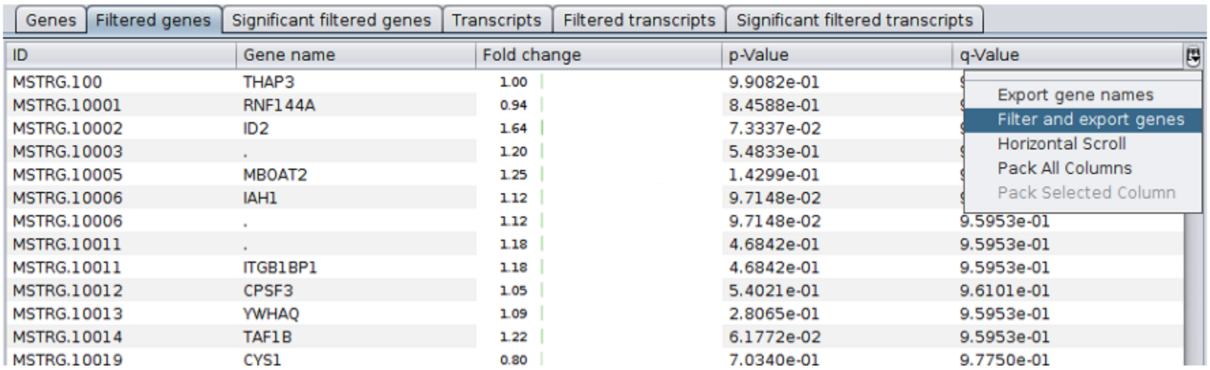
Output file :a/Analysis/test/analysis/ballgown.txt

Type of genes ● Overexpressed ○ Underexpressed

Number of genes 10

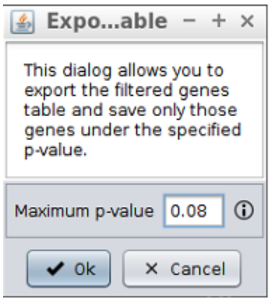
6.1.2.3 Creation of additional filtered genes tables

The *Filtered genes* table allows to generate the third analysis described in section *6.1.1 Ballgown outputs*. To perform this analysis, first the button over the vertical scroll must be clicked. Then, the *Filter and export genes* option must be selected. This will open a new window.



ID	Gene name	Fold change	p-Value	q-Value
MSTRG.100	THAP3	1.00	9.9082e-01	9
MSTRG.10001	RNF144A	0.94	8.4588e-01	9
MSTRG.10002	ID2	1.64	7.3337e-02	9
MSTRG.10003	.	1.20	5.4833e-01	9
MSTRG.10005	MBOAT2	1.25	1.4299e-01	9
MSTRG.10006	IAH1	1.12	9.7148e-02	9
MSTRG.10006	.	1.12	9.7148e-02	9.5953e-01
MSTRG.10011	.	1.18	4.6842e-01	9.5953e-01
MSTRG.10011	ITGB1BP1	1.18	4.6842e-01	9.5953e-01
MSTRG.10012	CPSF3	1.05	5.4021e-01	9.6101e-01
MSTRG.10013	YWHAQ	1.09	2.8065e-01	9.5953e-01
MSTRG.10014	TAF1B	1.22	6.1772e-02	9.5953e-01
MSTRG.10019	CYS1	0.80	7.0340e-01	9.7750e-01

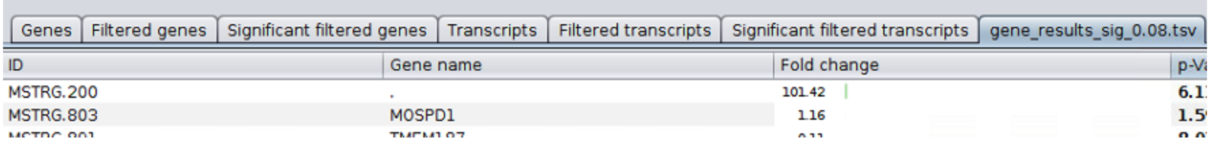
On this new window, DEWE will request for the *maximum p-value*, that specifies the maximum p-value of the genes in the table. After clicking the *Ok* button, the analysis will be generated in the selected path.



This dialog allows you to export the filtered genes table and save only those genes under the specified p-value.

Maximum p-value: 0.08

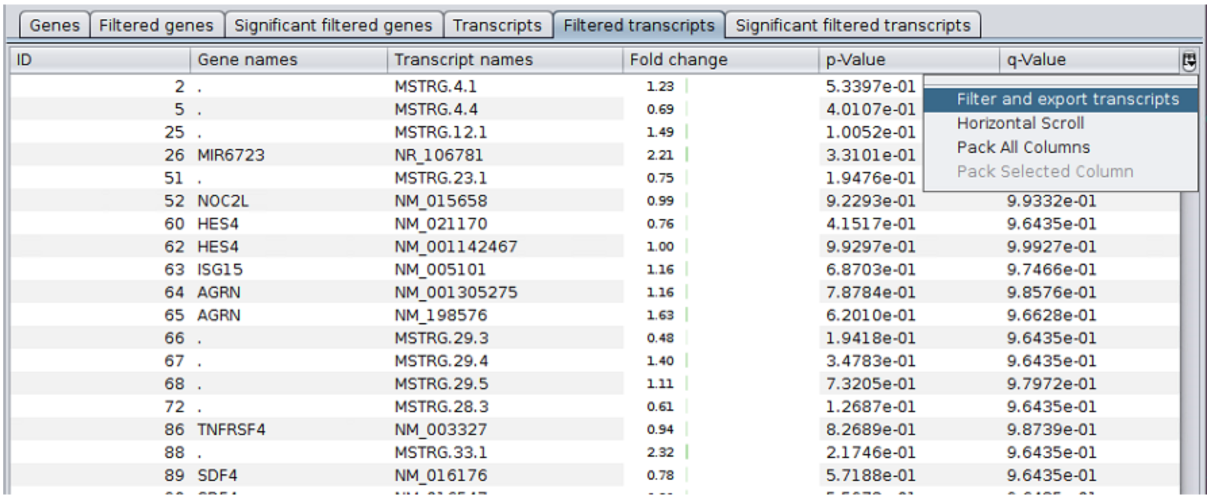
After the additional table is created, it will be displayed in the Ballgown table tabs.


Genes
Filtered genes
Significant filtered genes
Transcripts
Filtered transcripts
Significant filtered transcripts
gene_results_sig_0.08.tsv

ID	Gene name	Fold change	p-Va
MSTRG.200	.	101.42	6.1
MSTRG.803	MOSPD1	1.16	1.5

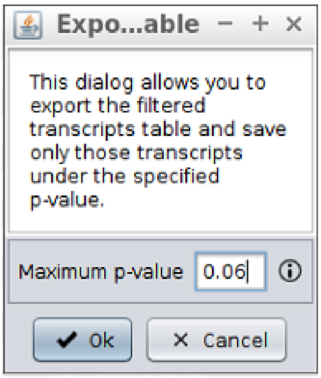
6.1.2.4 Creation of additional filtered transcripts tables

The *Filtered transcripts* table allows to generate the *fourth* analysis described in section *6.1.1 Ballgown outputs*. To perform this analysis, first the button over the vertical scroll must be clicked. Then, the *Filter and export transcripts* option must be selected. This will open a new window.



ID	Gene names	Transcript names	Fold change	p-Value	q-Value
2	.	MSTRG.4.1	1.23	5.3397e-01	
5	.	MSTRG.4.4	0.69	4.0107e-01	
25	.	MSTRG.12.1	1.49	1.0052e-01	
26	MIR6723	NR_106781	2.21	3.3101e-01	
51	.	MSTRG.23.1	0.75	1.9476e-01	
52	NOC2L	NM_015658	0.99	9.2293e-01	9.9332e-01
60	HES4	NM_021170	0.76	4.1517e-01	9.6435e-01
62	HES4	NM_001142467	1.00	9.9297e-01	9.9927e-01
63	ISG15	NM_005101	1.16	6.8703e-01	9.7466e-01
64	AGRN	NM_001305275	1.16	7.8784e-01	9.8576e-01
65	AGRN	NM_198576	1.63	6.2010e-01	9.6628e-01
66	.	MSTRG.29.3	0.48	1.9418e-01	9.6435e-01
67	.	MSTRG.29.4	1.40	3.4783e-01	9.6435e-01
68	.	MSTRG.29.5	1.11	7.3205e-01	9.7972e-01
72	.	MSTRG.28.3	0.61	1.2687e-01	9.6435e-01
86	TNFRSF4	NM_003327	0.94	8.2689e-01	9.8739e-01
88	.	MSTRG.33.1	2.32	2.1746e-01	9.6435e-01
89	SDF4	NM_016176	0.78	5.7188e-01	9.6435e-01

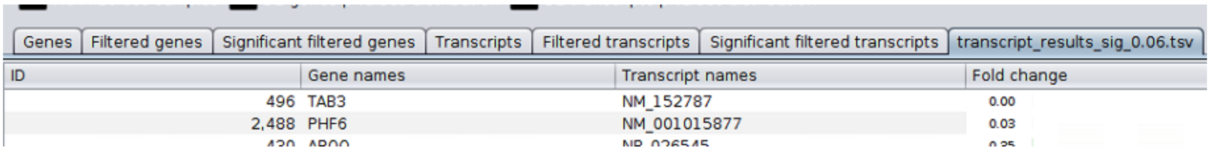
On this new window DEWE will request for *maximum p-value*, that specifies the maximum p-value of the transcripts in the table. After clicking the *Ok* button, the analysis will be generated in the selected path.



This dialog allows you to export the filtered transcripts table and save only those transcripts under the specified p-value.

Maximum p-value: 0.06

After the additional table is created, it will be displayed in the Ballgown table tabs.

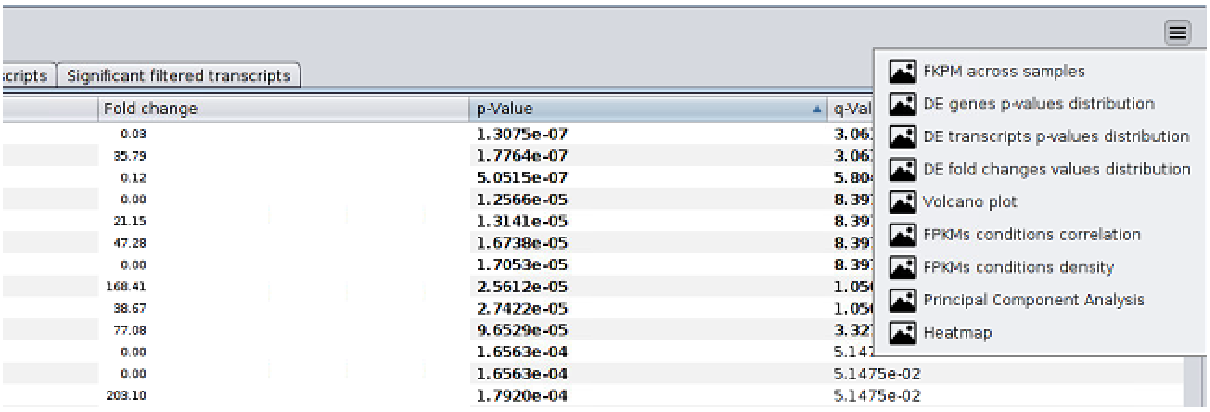

Genes
Filtered genes
Significant filtered genes
Transcripts
Filtered transcripts
Significant filtered transcripts
transcript_results_sig_0.06.tsv

ID	Gene names	Transcript names	Fold change
496	TAB3	NM_152787	0.00
2,488	PHF6	NM_001015877	0.03
430	APOO	NR_026545	0.35

6.1.2.5 Creation of colored figures

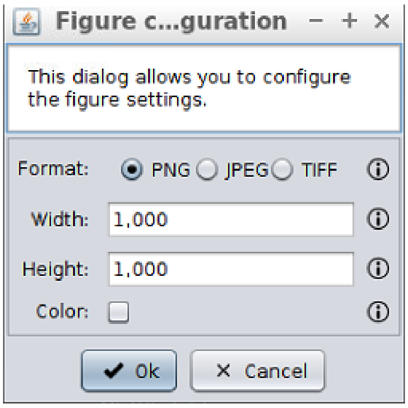
Over the tabs of the different tables generated, left aligned, there are three buttons that allow you to regenerate the following figures again:

- *FPKM across samples*: plot of the distribution of FPKM values across the samples.
- *genes DE pValues distribution*:. plot of the overall distribution of differential expression p-values for genes.
- *transcripts DE pValues distribution*: plot of the overall distribution of differential expression p-values for transcripts
- *Volcano plot:* combines the results of the p-values obtained after statistical tests with every single fold change.
- *Fold changes DE values distribution*: distribution of differential expression values histogram, which shows the frequency of the different expression fold-changes in a given experiment.
- *FPKM correlation plot:* FPKMs correlation between conditions, which are plotted in the x/y axis and will highlight those genes with enriched FPKM values in one or other experimental condition.
- *FPKM density plot:* FPKM correlation between conditions, represented as a density plot.
- *Heatmap:* heatmap of statistically significant genes (q-value<0.05), which a computed row and column clustering, the latter with an additional layer including the phenotype data.
- *Principal Component Analysis:* the global variance of the experiment is decomposed and grouped into new and orthogonal variables denominated components.


:cripts
Significant filtered transcripts
FKPM across samples
DE genes p-values distribution
DE transcripts p-values distribution
DE fold changes values distribution
Volcano plot
FPKMs conditions correlation
FPKMs conditions density
Principal Component Analysis
Heatmap

Fold change	p-Value	q-Val
0.03	1.3075e-07	3.06
35.79	1.7764e-07	3.06
0.12	5.0515e-07	5.80
0.00	1.2566e-05	8.39
21.15	1.3141e-05	8.39
47.28	1.6738e-05	8.39
0.00	1.7053e-05	8.39
168.41	2.5612e-05	1.05
38.67	2.7422e-05	1.05
77.08	9.6529e-05	3.32
0.00	1.6563e-04	5.14
0.00	1.6563e-04	5.1475e-02
203.10	1.7920e-04	5.1475e-02

Clicking on any of the options, DEWE will display a new window where the user will have to enter the format, resolution and color (colored or grayscale) of the figure that wants to generate. Once the OK button is clicked, the new figure will be generated inside the user-images folder, contained in the Ballgown results folder.



This dialog allows you to configure the figure settings.

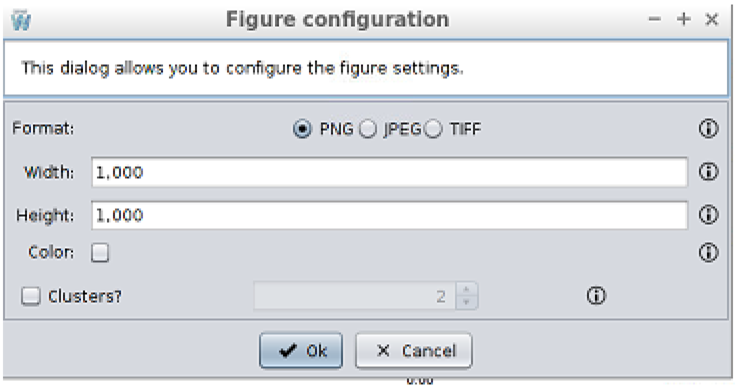
Format: ● PNG ○ JPEG ○ TIFF

Width: 1,000

Height: 1,000

Color: □

Additionally, in the creation of additional *Heatmap* plots, DEWE allows clustering the genes by a user-selected number of clusters.



This dialog allows you to configure the figure settings.

Format: ● PNG ○ JPEG ○ TIFF

Width: 1,000

Height: 1,000

Color: □

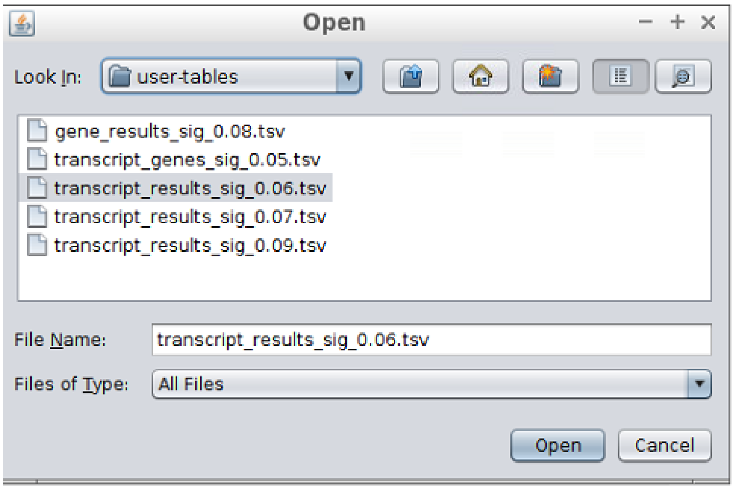
□ Clusters? 2

6.1.2.6 Visualisation of the additional filtered tables

Over the tabs of the different tables generated, right aligned, there are two buttons that allow you to load the previous generated additional filter tables for genes and transcripts:

- *Load genes*: Load a user-created filtered genes table.
- *Load transcripts*: Load a user-created filtered transcripts table.

Clicking on either option will open a new window where the user must select the table that he wants to import.



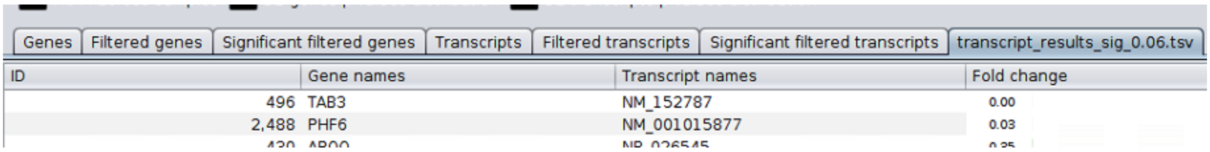
Look In: user-tables

gene_results_sig_0.08.tsv
transcript_genes_sig_0.05.tsv
transcript_results_sig_0.06.tsv
transcript_results_sig_0.07.tsv
transcript_results_sig_0.09.tsv

File Name: transcript_results_sig_0.06.tsv

Files of Type: All Files

After selecting the table and clicking on the OK button, if the table is correct it will be loaded in the Ballgown tables tabs.


Genes
Filtered genes
Significant filtered genes
Transcripts
Filtered transcripts
Significant filtered transcripts
transcript_results_sig_0.06.tsv

ID	Gene names	Transcript names	Fold change
496	TAB3	NM_152787	0.00
2,488	PHF6	NM_001015877	0.03
430	ARPO	NR_026545	0.35

6.2 edgeR

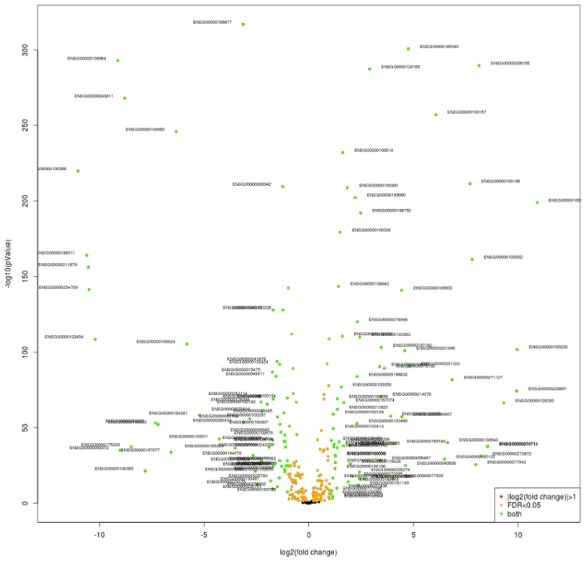
The edgeR differential expression analysis can be run from the menu operation (section 5.7.2) or as part of the available workflows (section 4.2 and 4.3). This section explains these outputs and the visualisation capabilities of the tool.

6.2.1 edgeR outputs

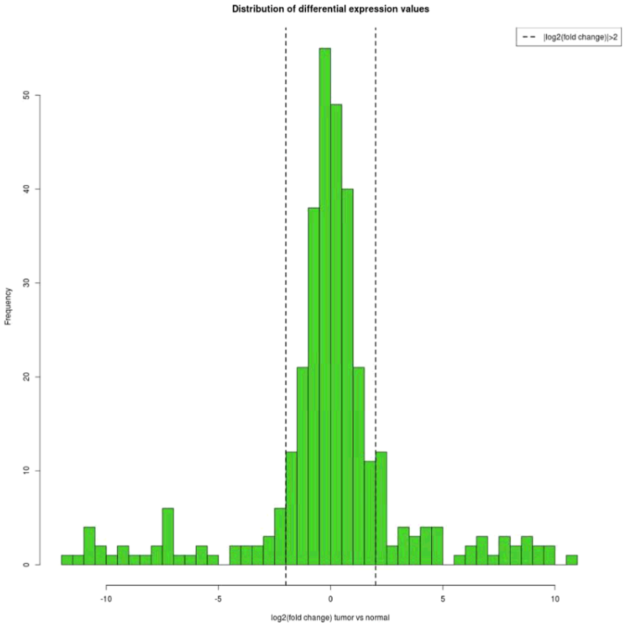
After performing a differential expression analysis with edgeR, the following single analysis are generated:

- *DE_genes.tsv*: differentially expressed genes between the two conditions.
- *DE_significant_genes.tsv*: significant differentially expressed genes between the two conditions, i.e. genes with p-value<0.05.
- *volcano-plot.jpeg*: This graphic combines the results of the p-values obtained after statistical tests with every single fold change. It will allow a rapid identification of the genes that increases both their magnitude of fold change and their statistical significance. Orange color represent up- or down-regulated genes (absolute value of log2fold-change > 1). Red color represent genes changing their expression with a statistical significance equal or lower than 0.05, measured as the adjusted p-value using the false discovery rate method. Green color represent up- or down-regulated genes with combine the two previous criteria. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color

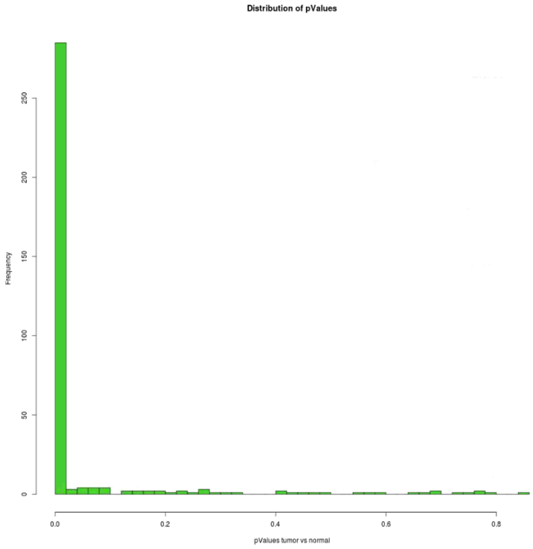
chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the edgeR results folder.



- *DE-values-distribution.jpeg*: distribution of differential expression values histogram, which shows the frequency of the different expression fold-changes in a given experiment. This will give the user an idea of the proportion of genes changing their expression over a given fold-change threshold, which is depicted with a dashed line. At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the edgeR results folder.



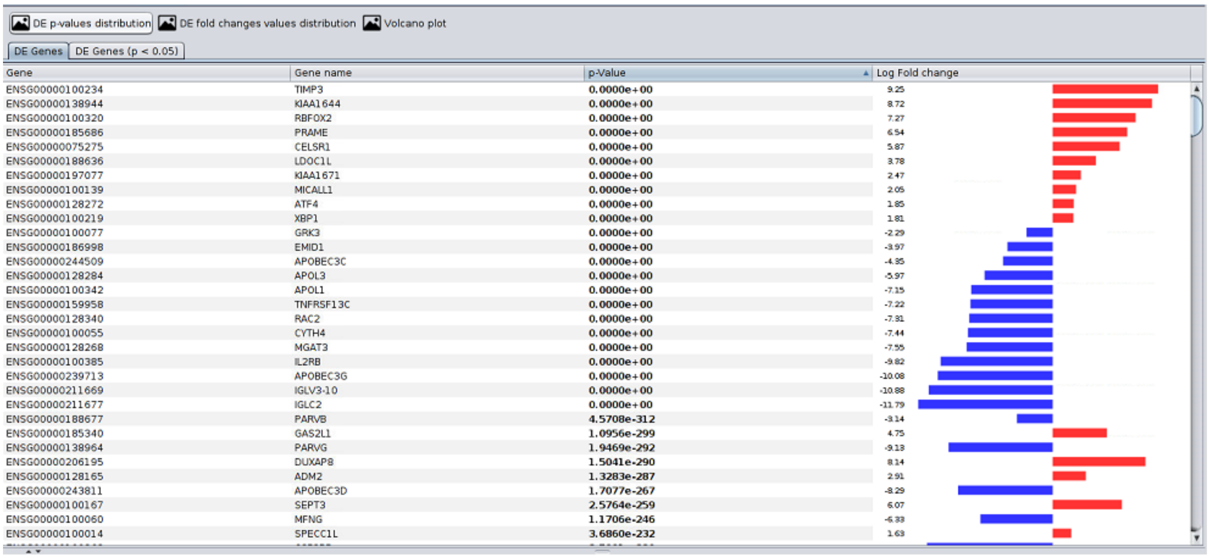
- *pValues-distribution.jpeg*: histogram of the frequencies of experimental p-values corresponding to the genes contained in the analysed genome. It will provide the user with a distribution of the differential expression p-values for all the genes, and it will also provide an indication of the subset of genes without significative differences in their fold-changes (p-value > 0.05). At first this image will be created in grayscale, but later it will be able to be generated again in the format, size and color chosen by the user (colored or grayscale). These user generated images will be saved in the *user-images* folder, contained within the edgeR results folder.



6.2.2 Results visualisation

The viewer enables the interactive browsing of genes analysis through a generated table to better understand and visualisation this analysis in the edgeR working directory.
As you can see in the following image, this view contains one tab:

- *DE Genes*: this tab contains a table with the genes in file *DE_genes.tsv.*
- *DE Genes (p<0.05)*: this tab contains a table with the genes in file *DE_significant_genes.tsv.*

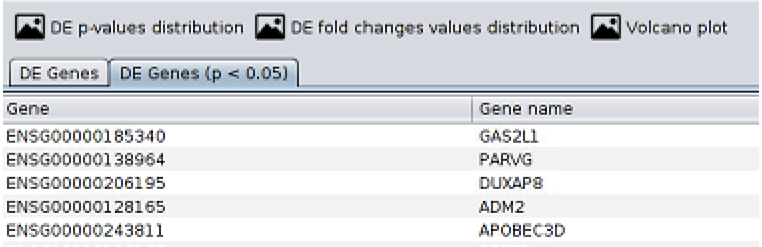


Gene	Gene name	p-Value	Log Fold change
ENSG00000100234	TIMP3	0.0000e+00	9.25
ENSG00000138944	KIAA1644	0.0000e+00	8.72
ENSG00000100320	RBFOX2	0.0000e+00	7.27
ENSG00000185686	PRAME	0.0000e+00	6.54
ENSG00000075275	CELSR1	0.0000e+00	5.87
ENSG00000188636	LDOC1L	0.0000e+00	3.78
ENSG00000197077	KIAA1671	0.0000e+00	2.47
ENSG00000100139	MICALL1	0.0000e+00	2.05
ENSG00000128272	ATF4	0.0000e+00	1.85
ENSG00000100219	XBP1	0.0000e+00	1.81
ENSG00000100077	GRK3	0.0000e+00	-2.29
ENSG00000186998	EMID1	0.0000e+00	-3.97
ENSG00000244509	APOBEC3C	0.0000e+00	-4.35
ENSG00000128284	APOL3	0.0000e+00	-5.97
ENSG00000100342	APOL1	0.0000e+00	-7.15
ENSG00000159958	TNFRSF13C	0.0000e+00	-7.22
ENSG00000128340	RAC2	0.0000e+00	-7.31
ENSG00000100055	CYTH4	0.0000e+00	-7.44
ENSG00000128268	MGAT3	0.0000e+00	-7.55
ENSG00000100385	IL2RB	0.0000e+00	-9.82
ENSG00000239713	APOBEC3G	0.0000e+00	-10.08
ENSG00000211669	IGLV3-10	0.0000e+00	-10.88
ENSG00000211677	IGLC2	0.0000e+00	-11.79
ENSG00000188677	PARVB	4.5708e-312	-3.14
ENSG00000185340	GAS2L1	1.0956e-299	4.75
ENSG00000138964	PARVG	1.9469e-292	-9.13
ENSG00000206195	DUXAP8	1.5041e-290	8.14
ENSG00000128165	ADM2	1.3283e-287	2.91
ENSG00000243811	APOBEC3D	1.7077e-267	-8.29
ENSG00000100167	SEPT3	2.5764e-259	6.07
ENSG00000100060	MFNG	1.1706e-246	-6.33
ENSG00000100014	SPECC1L	3.6860e-232	1.63

6.2.2.1 Creation of colored figures

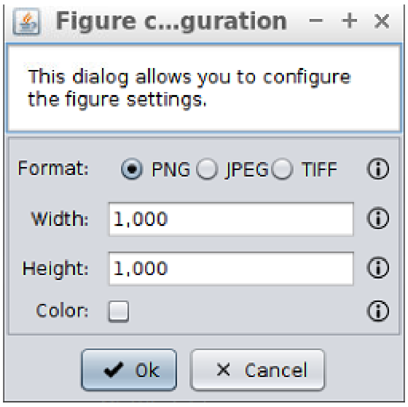
Over the tabs of the different tables generated, left aligned, there are three buttons that allow you to regenerate the following figures again:

- *genes DE p-values distribution*:. plot of the overall distribution of differential expression p-values for genes.
- *Fold changes DE values distribution*: distribution of differential expression values histogram, which shows the frequency of the different expression fold-changes in a given experiment.
- *Volcano plot:* combines the results of the p-values obtained after statistical tests with every single fold change.



Gene	Gene name
ENSG00000185340	GAS2L1
ENSG00000138964	PARVG
ENSG00000206195	DUXAP8
ENSG00000128165	ADM2
ENSG00000243811	APOBEC3D

Clicking on any of the options, DEWE will display a new window where the user will have to enter the format, resolution and color (colored or grayscale) of the figure that wants to generate. Once the OK button is clicked, the new figure will be generated inside the user-images folder, contained in the edgeR results folder.



This dialog allows you to configure the figure settings.

Format: ● PNG ○ JPEG ○ TIFF

Width: 1,000

Height: 1,000

Color: □

References

1. Dykes IM, Emanueli C. Transcriptional and Post-transcriptional Gene Regulation by Long Non-coding RNA. Genomics Proteomics Bioinformatics. 2017;15: 177–186.

2. Westermann AJ, Barquist L, Vogel J. Resolving host-pathogen interactions by dual RNA-seq. PLoS Pathog. 2017;13: e1006033.

3. Trapnell C, Williams BA, Pertea G, Mortazavi A, Kwan G, van Baren MJ, et al. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. Nat Biotechnol. 2010;28: 511–515.

4. Conesa A, Madrigal P, Tarazona S, Gomez-Cabrero D, Cervera A, McPherson A, et al. A survey of best practices for RNA-seq data analysis. Genome Biol. 2016;17: 13.

5. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. Nat Methods. 2012;9: 357–359.

6. Kim D, Langmead B, Salzberg SL. HISAT: a fast spliced aligner with low memory requirements. Nat Methods. 2015;12: 357–360.

7. Pertea M, Pertea GM, Antonescu CM, Chang T-C, Mendell JT, Salzberg SL. StringTie enables improved reconstruction of a transcriptome from RNA-seq reads. Nat Biotechnol. 2015;33: 290–295.

8. Trapnell C, Hendrickson DG, Sauvageau M, Goff L, Rinn JL, Pachter L. Differential analysis of gene regulation at transcript resolution with RNA-seq. Nat Biotechnol. 2012;31: 46–53.

9. Mezlini AM, Smith EJM, Fiume M, Buske O, Savich GL, Shah S, et al. iReckon: Simultaneous isoform discovery and abundance estimation from RNA-seq data. Genome Res. 2012;23: 519–529.

10. Frazee AC, Pertea G, Jaffe AE, Langmead B, Salzberg SL, Leek JT. Ballgown bridges the gap between transcriptome assembly and expression analysis. Nat Biotechnol. 2015;33: 243–246.

11. Robinson MD, McCarthy DJ, Smyth GK. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics. 2010;26: 139–140.

12. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2 [Internet]. 2014. doi:10.1101/002832

13. Hardcastle TJ. Generalized empirical Bayesian methods for discovery of differential data in high-throughput biology. Bioinformatics. 2015; btv569.

14. Seyednasrollah F, Laiho A, Elo LL. Comparison of software packages for detecting differential expression in RNA-seq studies. Brief Bioinform. 2015;16: 59–70.

15. Pertea M, Kim D, Pertea GM, Leek JT, Salzberg SL. Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown. Nat Protoc. 2016;11: 1650–1667.

16. Trapnell C, Roberts A, Goff L, Pertea G, Kim D, Kelley DR, et al. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. Nat Protoc. 2012;7: 562–578.

17. Grana O, Rubio-Camarillo M, Fdez-Riverola F, Pisano DG, Glez-Pena D. Nextpresso: Next Generation Sequencing Expression Analysis Pipeline. CBIO. 2017;12. doi:10.2174/1574893612666170810153850

18. Poplawski A, Marini F, Hess M, Zeller T, Mazur J, Binder H. Systematically evaluating interfaces for RNA-seq analysis from a life scientist perspective. Brief Bioinform. 2016;17: 213–223.

19. Delhomme N, Padioleau I, Furlong EE, Steinmetz LM. easyRNASeq: a bioconductor package for processing RNA-Seq data. Bioinformatics. 2012;28: 2532–2533.

20. Afgan E, Baker D, van den Beek M, Blankenberg D, Bouvier D, Čech M, et al. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. Nucleic Acids Res. 2016;44: W3–W10.

21. Russo F, Righelli D, Angelini C. Advancements in RNASeqGUI towards a Reproducible Analysis of RNA-Seq Experiments. Biomed Res Int. 2016;2016: 7972351.

22. Lohse M, Bolger AM, Nagel A, Fernie AR, Lunn JE, Stitt M, et al. RobiNA: a user-friendly, integrated software solution for RNA-Seq-based transcriptomics. Nucleic Acids Res. 2012;40: W622–W627.

23. Wang Y, Mehta G, Mayani R, Lu J, Souaiaia T, Chen Y, et al. RseqFlow: workflows for RNA-Seq data analysis. Bioinformatics. 2011;27: 2598–2600.

24. Icay K, Chen P, Cervera A, Rantanen V, Lehtonen R, Hautaniemi S. SePIA: RNA and small RNA sequence processing, integration, and analysis. BioData Min. 2016;9: 20.

25. da Veiga Leprevost F, Grüning BA, Alves Aflitos S, Röst HL, Uszkoreit J, Barsnes H, et al. BioContainers: an open-source and community-driven framework for software standardization. Bioinformatics. 2017;33: 2580–2582.

26. Griffith M, Walker JR, Spies NC, Ainscough BJ, Griffith OL. Informatics for RNA Sequencing: A Web Resource for Analysis on the Cloud. PLoS Comput Biol. 2015;11: e1004393.