Software installation guide for whole genome sequencing and transposon mapping

Contents

Introduction

This file discusses the installation procedures for the programs required for whole genome sequencing analysis and transposon mapping. The numbering corresponds with the order in which this software is used. Note that a part of the software is for windows and Linux, and a part is for Linux only. Therefore a virtual machine is included to run Ubuntu (or another unix based operating system to your liking). For a more detailed workflow and the commands of all the individual commands, see satay_analysis_notes.md.

A Linux vrtual machine is already created that is completely set up and includes all the software discussed below. This is located on the N-drive in the 'VirtualMacines' folder. Step 0 below discusses how this virtual machine can be installed on your computer and how to use it. The following steps are only important when you want to create a virtual machine from scratch so you can set it up completely to your liking or using another operating system than Ubuntu that is used existing virtual machine on the N-drive.

To run a program in the command line, you need to specify the entire path to the location where the program is installed. An easier way is to add the program to the system path. This can be done with the command `setx PATH '/path/to/program'`. To check if the program is added to the path, use `echo $PATH$`.

Some programs requires java to be installed. To check if java is installed, run the command 'java -version' in the command line. If it is not installed, download and install it from https://www.java.com/nl/download/

Using the already existing virtual machine

0. Setting up already existing virtual machine

https://www.virtualbox.org/

Running a virtual machine requires VirtualBox (VB), which can be downloaded from the above link and installed using the normal procedure for installing software on Windows machines. The memory of the virtual machine (called the virtual hard drive (vhd)) is stored on the N-drive in the VirtualMachines folder with the name VM_Ubuntu64_1.vhd. **Before using this virtual machine, make sure you make a copy of the virtual hard drive** either to your local machine or, when there is not enough memory on your computer, somewhere on the drives in your personal folder. The file is over 50Gb in size, so it might be troublesome to store it locally. When storing it on the drives, make sure it has a stable internet connection when you use it since when the connection is lost, the virtual machine crashes as it cannot access its memory anymore.

To use the existing vhd, a new virtual machine has to be created within VirtualBox. To do this, apply the following steps.

1. Open VB -> click 'New' (blue icon)
2. Choose a name for the virtual machine and set 'type': Linux, 'Version': Ubuntu 64-bit.
3. Allocate RAM memory (adviced is to use the recommended amount to prevent problems or not more than 1/4 of the total memory available in your computer)
4. Select 'Use an existing virtual hard disk file' and use the yellow folder icon to search for your copied virtual hard drive.

The Virtual machine should now be working. Run the virtual machine by selecting 'Start'. It should start up and work like a normal Linux machine. For some things a password and username might be required, which can be found in the text file located in the same drive as the vhd file.

It might be good to make sure the system is updated. For this, apply the following steps:

1. Open the terminal app
2. Enter `sudo apt update` (requires password)
3. Enter `sudo apt upgrade`. Type 'Y' if the terminal asks if software needs to be installed.
4. Restart the system.

Things that might not be working yet are the guest additions. These are needed for optimizing the virtual machines performance and are needed for allowing the virtual machine to communicate with some of the hardware of the physical machine like screen size and shared folders (for example, if you use VirtualBox for first time, the screen of the virtual machine might be tiny, but this is because it does not yet recognize the screen size of your physical computer).

To set up the guest additions use the following commands and options

1. Enter `sudo apt install build-essential dkms linux-headers-$(uname -r)`
2. Click the 'devices' tab in the top of the window. Click 'Insert Guest Additions CD Image' and run the installer. Power off the VM when done.
3. In VB click 'settings' (yellow gear symbol) and go to the 'General' tab; 'Advanced'. Set both 'shared clipboard' and 'drag n drop' to bidirectional and click 'ok'
4. In VB click 'settings' again (yellow gear symbol) and go to the 'Shared Folders'. Add path to a folder that is going to be used as shared folder. Set 'Auto Mount' and click 'Ok'
5. Start the VM and check if the shared folder is present (either on the desktop, the files folder or in the media folder).
6. To automatically resize the screen of the virtual machine, go to the 'view' tab in the top of the screen and select 'Auto-resize guest display'.

The desktop of the virtual machine include the shared folder and a README file. The home directory of the VM (which is the default location where the terminal app opens from which all the commands are entered) includes the file processing_workflow.sh which is a workflow that automatically performs all the processing steps. In the files app, the documents folder includes three subfolders:

- data_processing: This location can be used to temporarily store the sequencing data.
- Reference_Sequences: This includes by default the reference sequences for the S288c and the W303 strain in .fasta format including some additional files that are needed for the alignment software.
- Software: This includes all the software packages required for the preprocessing and also includes the java download.

The adapters.fasta file in which the adapter sequences are stored that are used during trimming is located in ~/Documents/Software/BBMap/bbmap/resources/adapters.fa. Documents that are placed in the shared folder can be seen in both the virtual machine and the host computer. For a more detailed explanation of the Virtual Machine layout and how to use the workflow_processing.sh file, see the README file on the desktop of the VM.

Creating a virtual machine from scratch

1. VirtualBox-6.1.4

https://www.virtualbox.org/, https://ubuntu.com/download/desktop

VirtualBox is used for running Linux based sofware on a Windows machine. For the operating system, recommended is to download Ubuntu 18.04 LTS 64-bit from https://ubuntu.com/#download, but any other Linux OS should be fine. Save the downloaded document at a convenient location on your computer. To install:

1. Run the VirtualBox-6.1.4-136177-Win application to install Virtualbox (VB) on your computer
2. Open VB -> click 'New' (blue icon)
3. Choose a name for the virtual machine, type Linux, and Version Ubuntu 64-bit.
4. Allocate RAM memory (adviced is to use the recommended amount to prevent problems or not more than 1/4 of the total memory available in your computer)
5. Make new virtual hard drive (choose VHD)
6. Choose size of VHD (recommended is a minimal amount 50GB)
7. Go to settings of your new virtual machine (VM) (yellow gear symbol) and go to the storage tab.
8. Click 'empty' under 'Controller IDE'. Next to 'IDE Secondary master' click the blue CD icon and select 'choose/create a virtual optical disk'. Click 'Add' and choose your downloaded Linux OS.
9. Start the VM and run the Linux install process with the recommended settings. Note that this might freeze sometimes, so it might be necessary to reinstall it a few times)

The following steps are to add a shared folder between the Windows machine and the VM that allows for easier sharing of data (this is not obligatory). This requires to install 'guest additions' in the VM. If a folder is already shared and a new folder is wanted to be shared as well, perform only step 16, 17 and 18.

10. Open the terminal app
11. Enter `sudo apt update`
12. Enter `sudo apt upgrade`. Type 'Y' if the terminal asks if software needs to be installed. Restart the system.
13. Enter `sudo apt install build-essential dkms linux-headers-$(uname -r)`
14. Click the 'devices' tab in the top of the window. Click 'Insert Guest Additions CD Image' and run the installer. Power off the VM when done.
15. In VB click 'settings' (yellow gear symbol) and go to the 'General' tab; 'Advanced'. Set both 'shared clipboard' and 'drag n drop' to bidirectional and click 'ok'
16. In VB click 'settings' again (yellow gear symbol) and go to the 'Shared Folders'. Add path to a folder that is going to be used as shared folder. Set 'Auto Mount' and click 'Ok'
17. Start the VM and check if the shared folder is present (either on the desktop, the files folder or in the media folder).
18. To be able to access the folder, special permission needs to be given. For this open the Terminal app.
19. Enter `sudo adduser [username] vboxsf` (where [username] should be replaced with your actual username of the VM).

20. To check, enter `id [username]`. This should give a list that needs to include 'vboxsf'
21. Restart the VM.
22. To automatically resize the screen of the virtual machine, go to the 'view' tab in the top of the screen and select 'Auto-resize guest display'.

In order to run java based programs in Linux, Java needs to be installed. There are several ways of installing Java, but one of the simplest ways is using the command line with the following commands:

1. Enter `sudo -s` (enter password if requisted)
2. Enter `apt-get install openjdf-11-jre`. (If this version of java is not found, check which versions are available by entering `apt-get install openjdf` followed by a double tab. This should give a list of all available installations. It also gives you the opportunity to install the jdk (Java Developer Kit) instead of the jre (Java Runtime Environment), but if you don't plan to develop software in Java, the jre, what is used here, is good enough).
3. Restart the terminal. Check the installation by entering `java -version`

2. Fastqc (Windows or Linux)

https://www.bioinformatics.babraham.ac.uk/projects/fastqc/

Fastqc is used for quality checking. To install Fastqc in Linux, open the bash and go to the fastqc folder in the bash. Type in the command `sudo apt install fastqc` (this might require a password). To run interactively, click the 'run-fastqc' batch file. To run non-interactively, enter in 'fastqc' in the command line.

3a. Trimmomatic-0.39 (Windows or Linux)

http://www.usadellab.org/cms/?page=trimmomatic

Trimmomatic is used for trimming fastq files. It is java based and therefore does not need to be installed. To run, enter 'trimmomatic-0.39' in the command line. The adapters folder contains some adapters that can be used during trimming if desired.

123Fastq-v1.1 (Windows) (optional) https://sourceforge.net/projects/project-123ngs/

123Fastq is Fastqc and Trimmomatic combined in one interactive program. It is java based and therefore does not need to be installed. Click the 123fastq executable jar file to run the program.

3b. BBDuk-38.84 (Windows or Linux)

https://jgi.doe.gov/data-and-tools/bbtools/

BBDuk is an alternative for Trimmomatic for trimming of fastq files. It java based and therefore does not need to be installed. It is part of the bbtools packages (named the bbmap when downloaded). Once downloaded, unpack the .tar.gz package. Run the bbduk.sh executable in the bbmap directory. The adapter.fa file is included and located in the /resources directory.

4. BWA (Linux)

http://bio-bwa.sourceforge.net/

BWA is used for aligning the reads to a reference genome and to index the reference sequence. After downloading the software in the VM, install it by entering the following commands in the terminal:

```
bunzip2 bwa-0.7.17.tar.bz2
tar -xvf bwa-0.7.17.tar
cd bwa-0.7.17
sudo apt-get update
sudo apt-get install bwa
```

To run, enter `bwa` in the terminal.

5. SAMTools and bcftools (Linux)

http://www.htslib.org/

Samtools is used for processing after alignment, for example for converting SAM files to BAM files. After downloading the software in the VM, install it by entering the following commands in the terminal:

```
bunzip2 samtools-1.10.tar.bz2
tar -xvf samtools-1.10.tar
cd samtools-1.10
sudo apt-get update
sudo apt-get install samtools
```

To run, enter `samtools` in the terminal. Do the same protocol for bcftools.

6. Sambamba (Linux)

https://lomereiter.github.io/sambamba/

Sambamba is used for processing after alignment, for example for sorting and indexing the BAM files. After downloading the software in the VM, install it by entering the following commands in the terminal:

```
gunzip sambamba-0.7.1-linux-static.gz
chmod +x sambamba-0.7.1-linux-static
sudo ln -s /path/to/sambamba-0.7.1-linux-static /usr/local/bin
```

(where in the last line `/path/to/` needs to be replaced with the actual path.) To run, enter `sambamba` in the terminal.

7. IGV (Windows) (optional)

https://software.broadinstitute.org/software/igv/

IGV (Integrative Genomic Viewer) is used for visually check the results. Click the IGV_Win_2.8.0-installer and run the install process.

8. Matlab Transposon count (Windows)

https://sites.google.com/site/satayusers/complete-protocol/bioinformatics-analysis

This code relates the number of reads and transposon counts to the genes. This code is provided from the Kornmann-Lab.

9. Removing unused software

The installed version of Ubuntu comes with different application by default, for example an email application (thunderbird), an alternative to the Microsoft office package (LibreOffice) and some games. These application you will most likely not use and to save memory on your Virtual Machine, these can be uninstalled. To do this, go to the 'Ubuntu Software' application and go the tab 'installed'. Uninstall everything here that you are not planning to use. You can (re)install application here as well.

Installing additional software in Virtual Machine

10. upgrading Python

By default, Linux comes with Python already installed. In the terminal application, run the command `python --version` or `python3 --version` to see which version is installed. If python3 is installed with version 3.6, then this should be updated to python 3.7 or above. This can be done using the procedure explained on this website. Running the command `python3.7` should open python 3.7 instead of version 3.6.

Note that it might happen that the terminal does not launch anymore after upgrading the python version. This is something to do with the not properly removing one of the installation packages. To sort this out, go to the 'Ubuntu Sofware' application and install 'xterm'. When opening xterm, enter the commands `sudo rm /usr/bin/python3` followed by `sudo ln -s python3.6 /usr/bin/python3`. After this, close xterm and terminal should now launch again.

11. Getting miniconda for install python packages

The default Python application is quite bare equiped (for example it doesn't include numpy or matplotlib). These packages can be installed using a package installer. Two commonly used ones are pip and conda. Here we use conda. Conda is part of Anaconda, which is a distribution for Python and R. Installing Anaconda gives you a graphical user interface and many preinstalled packages, but this takes up a lot of space and many of these packages you probably do not need. To save memory on the virtual machine, install the minimal version of Anaconda called Miniconda.

1. Download the Linux installer for Miniconda from the conda website (download the Linux 64 bit version).
2. Check the hash code from the download by typing in the command `sha256sum [path/to/Miniconda3-latest-Linux-` (where the [path/to/Miniconda3-latest-Linux-x86_64.sh] needs to be replaced with the correct path and filename). This hash code should match exactly the has code provided on the download website.
3. If the hash code is correct, run the command `bash [path/to/Miniconda3-latest-Linux-x86_64.sh]`. Accept the license agreement and enter the location where the software needs to be installed (or leave this at the default location). When the installer prompts `Do you wish the installer to initialize Miniconda3 by running conda init`, enter `yes`.
4. Restart the terminal.
5. Check the installation by running the command `conda list`, which should give you a list of packages that are installed in Python.

To add a package run the command `conda install [package]`, where [package] should be replaced by the name of the package that you want to install. Recommended to install at least numpy and matplotlib. Conda uses specific locations to search for packages called channels. If you want some specific bioinformatic tools, it might be necessary to add the channel 'Bioconda' to Miniconda. To do this run the command `conda config --add channels bioconda`. Now, more packages can be installed, for example pysam to read bam files in python.