Welcome to the ASOs Design Tool developer guide. This tool is powered by R Studio Server and runs on a Linux server, providing a reliable and efficient environment for your design tasks. This guide will introduce you to the tool's features and how to use them effectively in our server setup.

This tool is for identifying target regions for antisense oligonucleotides. The script processes genetic sequences, predicts RNA accessibility, determines binding dynamics, and evaluates variants within the human genome. It applies filtering and clustering analyses to efficiently select potential target areas for therapeutic or research purposes.

Table of contents

[Working environment](#_30j0zll)

[How to use the tool](#_1fob9te)

[How to use the GUI version](#_hoxqehzb0fs9)

[How do I install the software and its dependencies](#_2et92p0)

[Common problems](#_tyjcwt)

[Can't open plfold\_dp.ps for dot plot](#_ir797vdto4ll)

[Running on a non-Linux operating system.](#_lc4cd99opfji)

[How to update the databases](#_1t3h5sf)

[Updating the databases can be a little more difficult.](#_4d34og8)

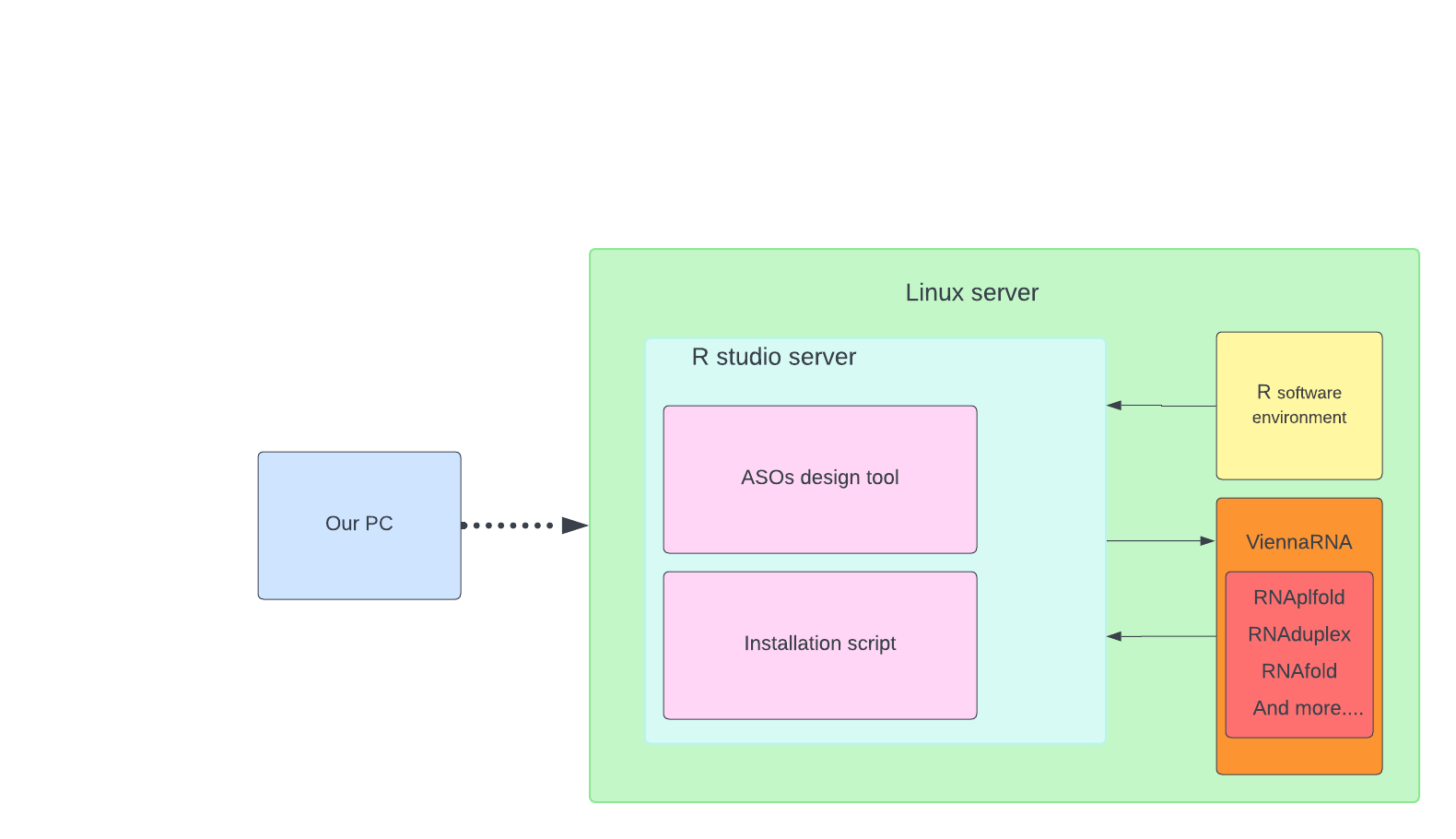
[Updating the Ensembl version](#_2s8eyo1)

Results example

[FAQ](#_17dp8vu)

# 

# Working environment

A small diagram of how this works is shown as follows:  


In our configuration, we utilize a Linux server that is connected to our windows PC via SSH.

SSH, which stands for Secure Shell, is a network protocol that provides administrators with a secure way to access a remote computer.

This server hosts the R Studio Server, which contains two key components: the "ASOs Design Tool", a specially developed R script for creating ASOs, and the "Installation Script", which includes multiple scripts for updates and necessary installations.

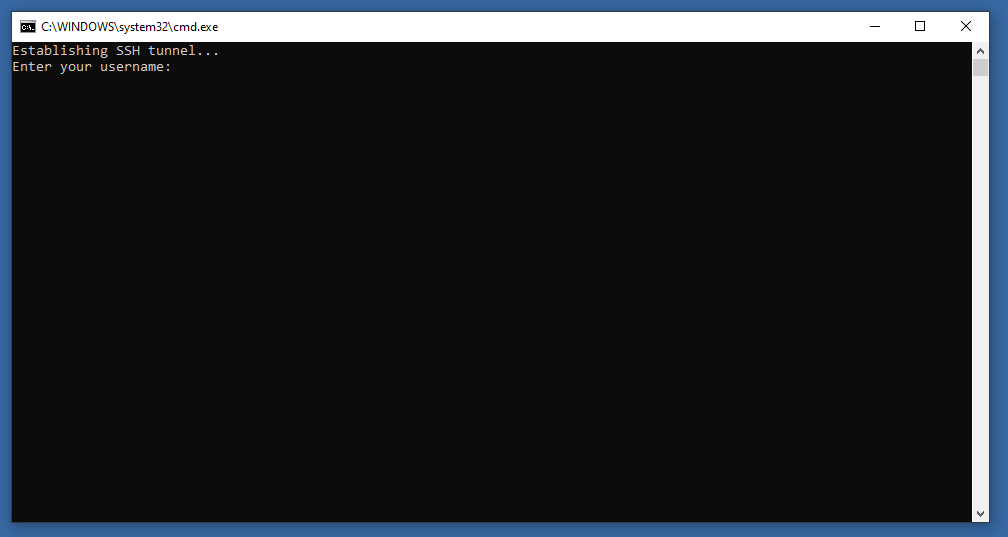
The operation of the R Studio Server depends on the "R Software Environment".

Additionally, we have integrated ViennaRNA, an external tool used for predicting RNA structures. This tool is directly accessible from the R Studio Server and provides various functionalities for our research purposes.

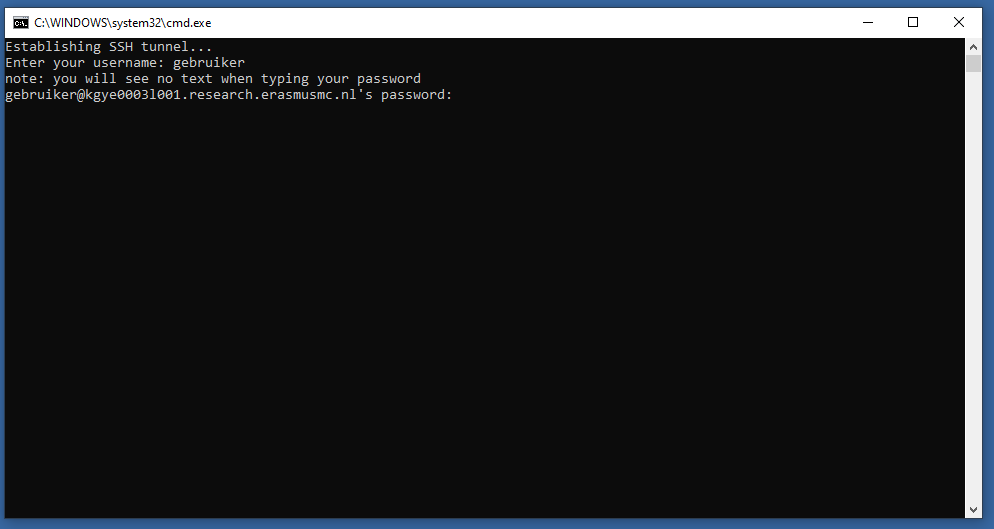
# How to use the tool

1)First, we need to connect to the Linux server. This is done with an SSH connection using the “connect to Rstudio.bat”

  
2)After running, an Internet browser tab and a black screen will open (this is the cmd window), here we will have to do a couple of steps so that we can connect to the server.

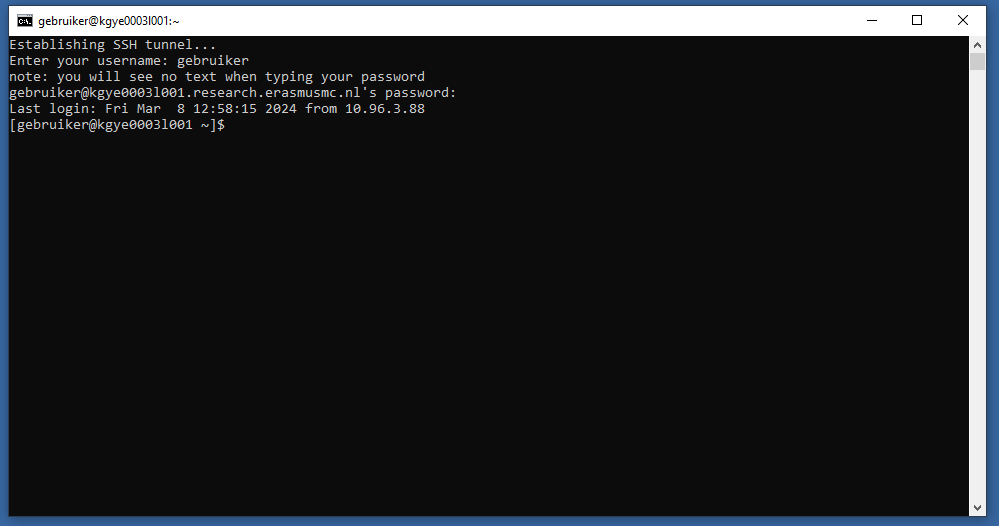


3)Enter the username “gebruiker” (note: some users with server access can use their microsection number and own password)

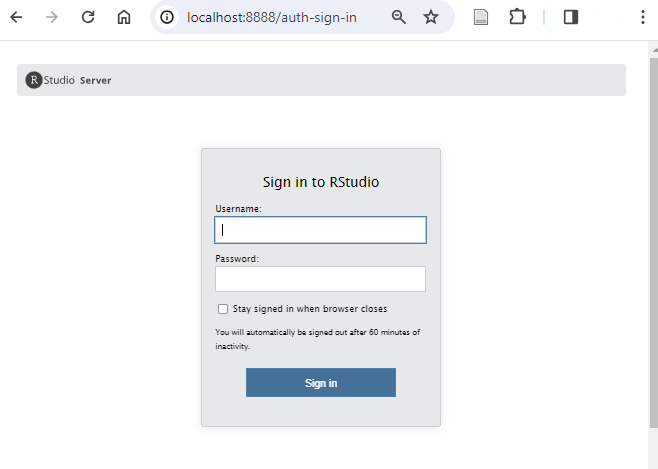
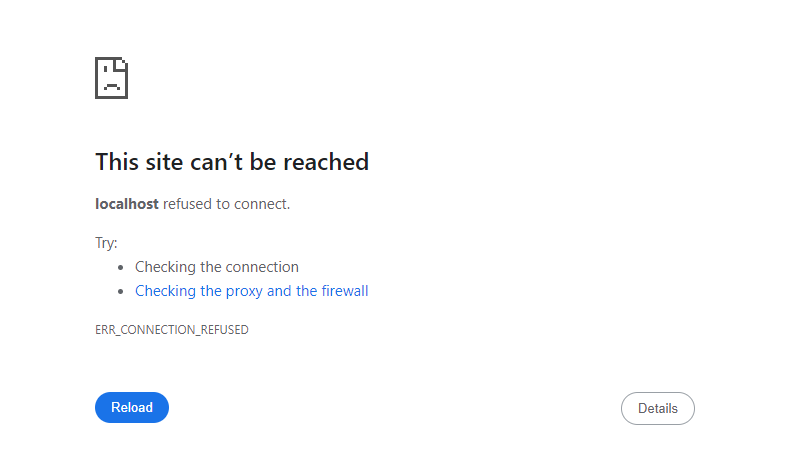
  
4)Now enter the password “admin”, you will not be able to see the password while typing (note: this is not an actual admin account”)

4.1) If you’re connecting to a new computer, you might get asked if you want to continue with a new fingerprint, just continue with “yes”

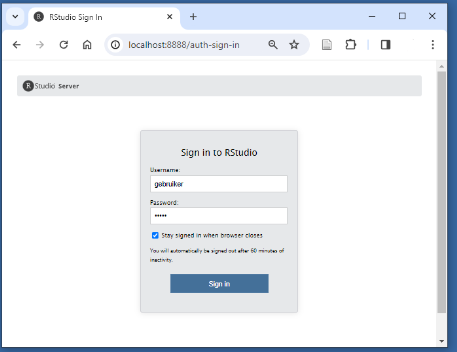


  
5)if you see this screen, then it means that you successfully connected with the Linux server, **DO NOT close this tab.** Closing this tab will stop the connection and kick you out of R Studio.

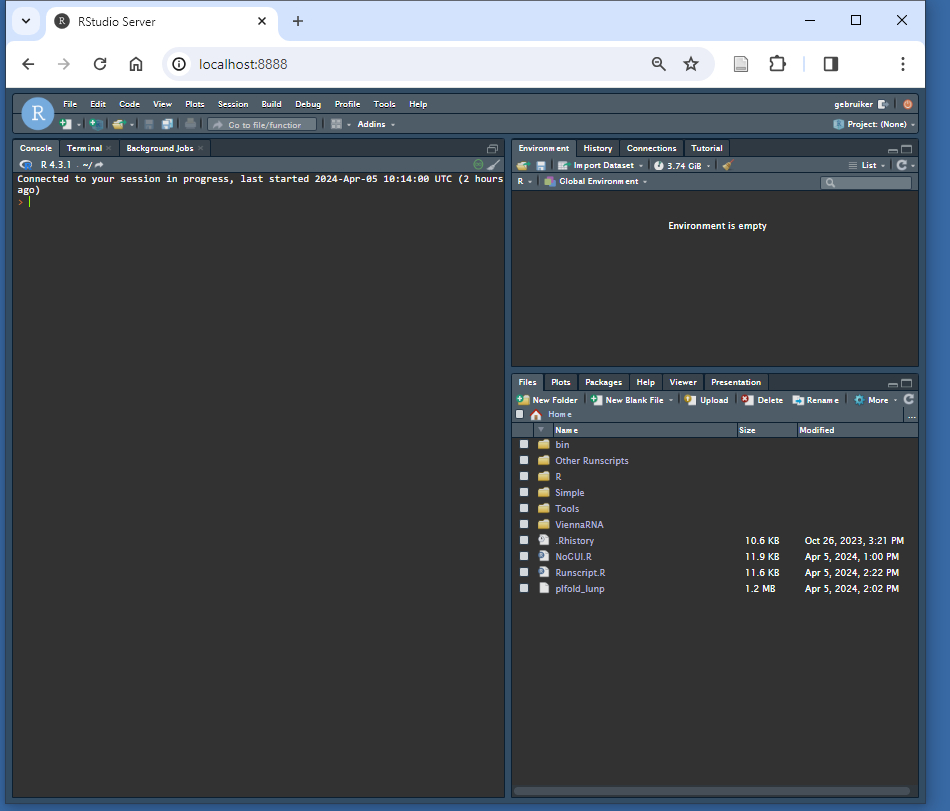
6)Now go to the internet browser tab opened by the bat file.

  
6.1) If you see this screen, try to refresh the page after login in at stap 3)  


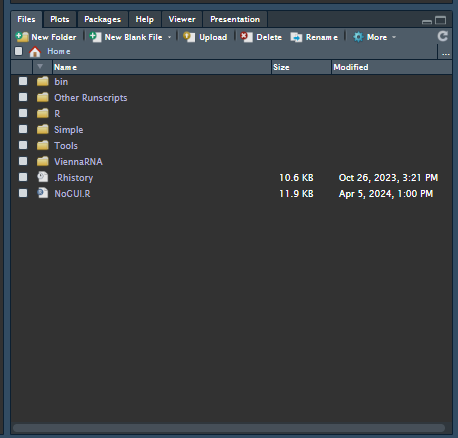
7)This is the login screen of the R studio server, here fill in the same username and password as before.



8)Press the Sign-in button.



9)On the bottom right you can find the files

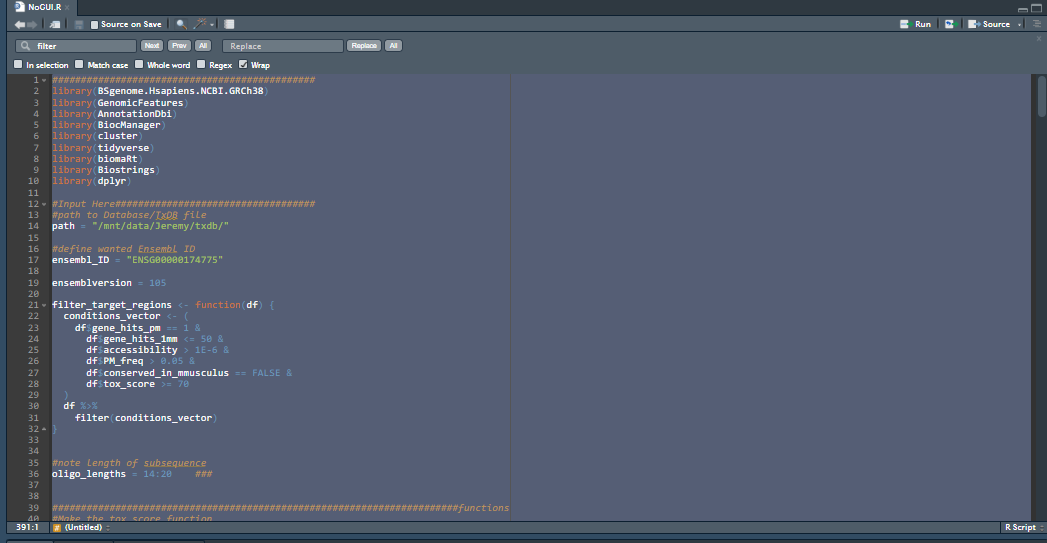


10)Now you can select which script you want, there are a couple of options

* NoGUI, this is simply the tool without the GUI.
* Simple, in this folder, you will find 2 files, it doesn't matter which one you use to run it.
  + In an R Shiny application, ui.R is responsible for defining the user interface's layout and appearance, while server.R handles the application's backend logic, processing user inputs from the UI to produce and display outputs.
* Tools, in this folder you can find the script for updating the database file, installing the packages and comparing CSV/results files.

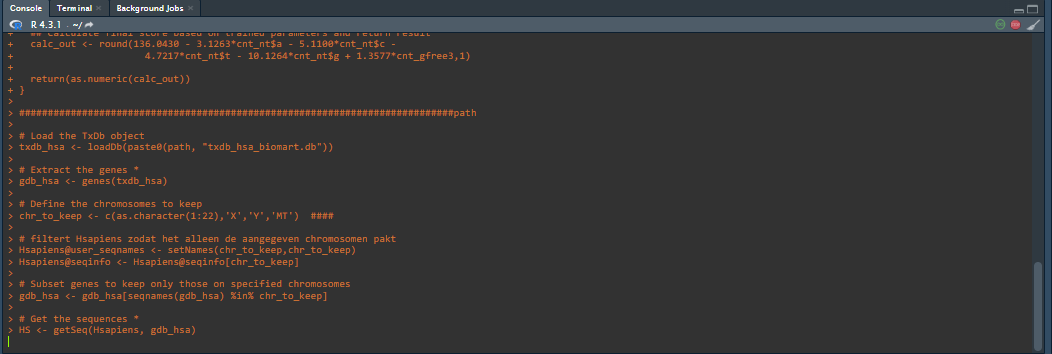
In this guide we shall be using the NoGUI.R as example.

11)open the script and select all the lines using the button combination: “Ctrl+A” on Windows/linux and “Commmand-A” on Mac.



12)Press Ctrl-enter and run all the lines

The console would look something like this

  
13) after the script is done, you shall see 2 files in the list of scripts with a name familiar to:

“Results\_output\_clustered\_2024-04-05 14:02:24.csv” **\*the date and time will differ**

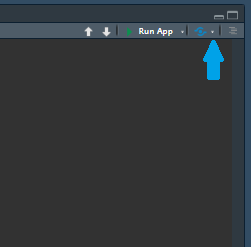
# 

# How to use the GUI version

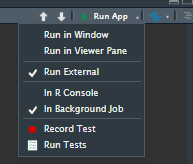
Follow steps 1 to 10 from the “How to use the tool"

1)Select the ui.R or server.R file and open it.

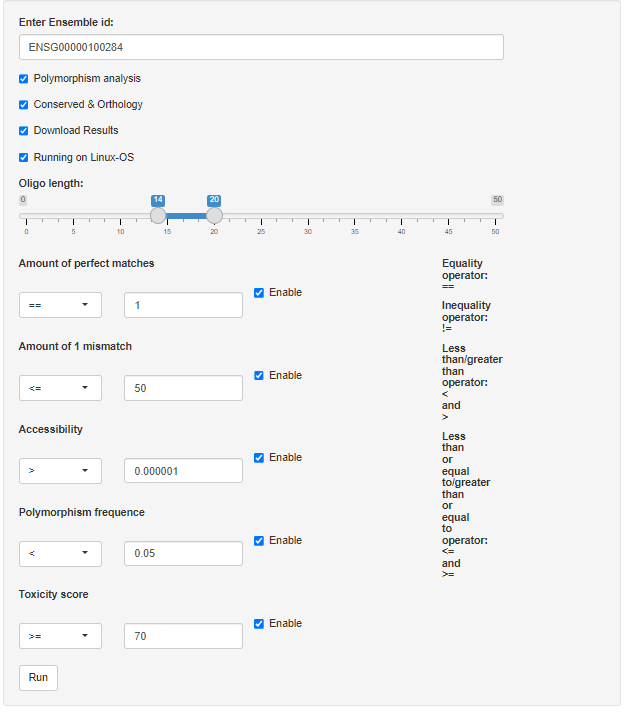
2)Press the small white arrow on the right of the “Run App” button.



3)Select “Run External” and “in Background Job”

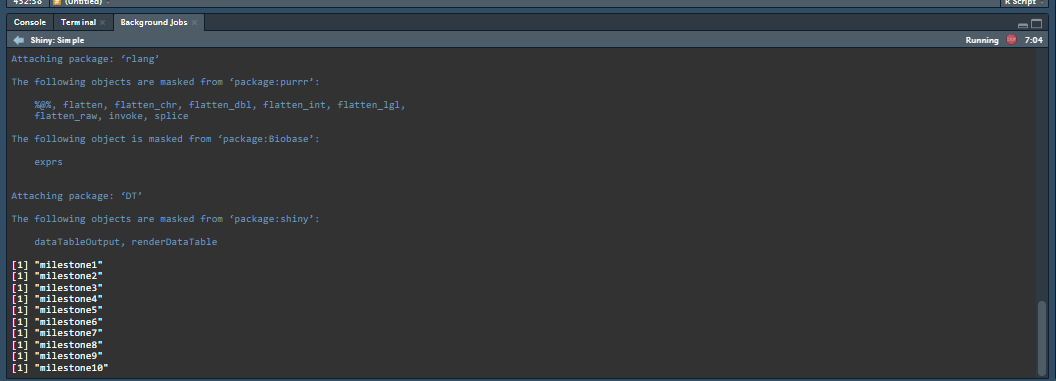


4) Now press run, a new window will open.



Here you can change the settings and parameters as wanted.

5)Press the run button, to check if it's running you could check the console for “milestone” prints.



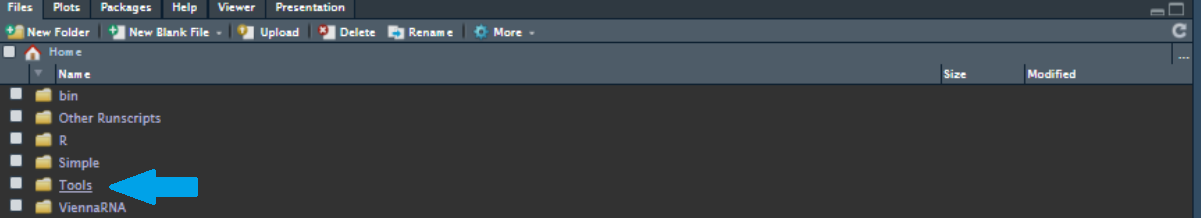
6)Here you can also see when it has finished, the results will be downloadable but also visible on the GUI.

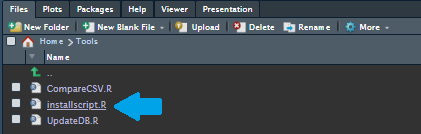
# How do I install the software and its dependencies

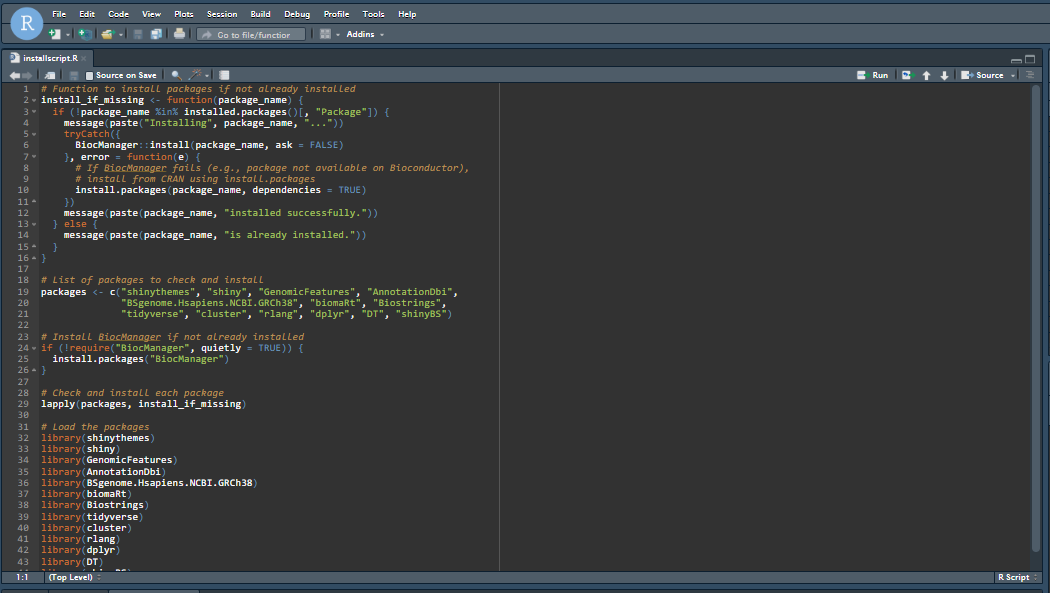
**Updating the packages is not necessary to keep the tool working, so only use this when needed!**

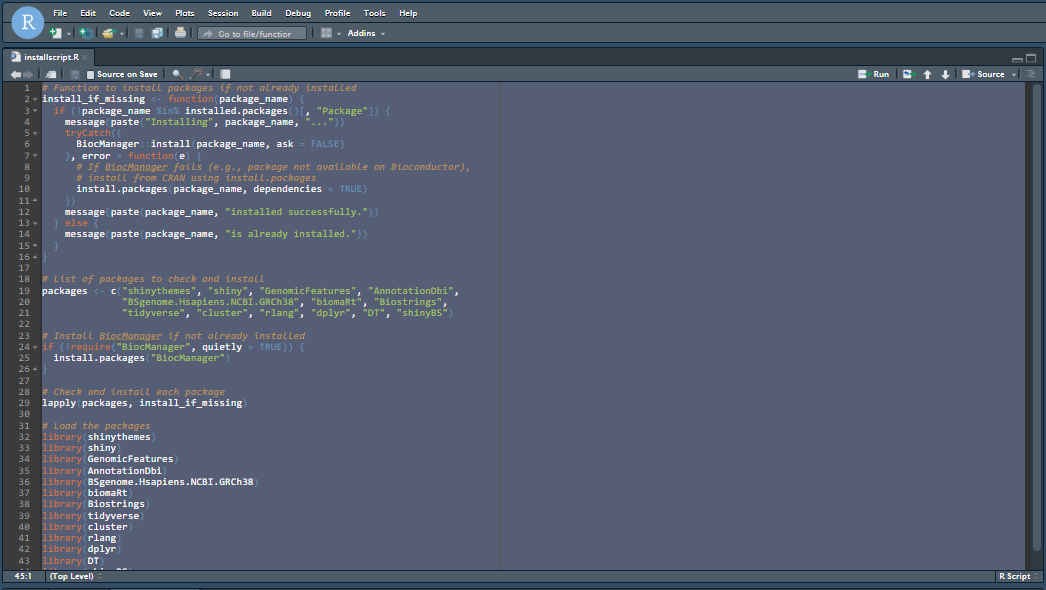
Installing and updating the dependencies can be done with the “installscript.R” inside the Tools folder which can used inside of the R studio environment:

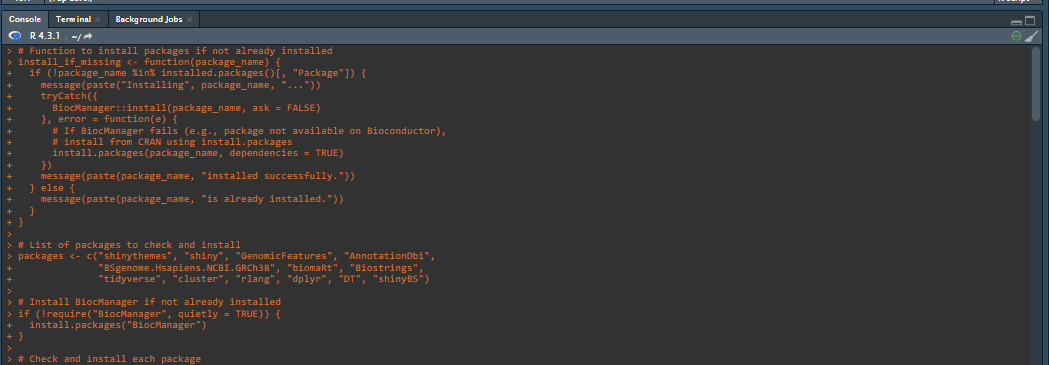
1) Simply open the script in the lower right.



  
2)When the script is opened select all the lines using the button combination: “Ctrl+A” on Windows/linux and “Commmand-A” on Mac.



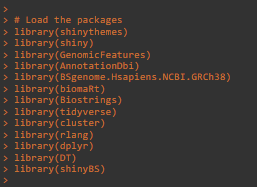
  
3)Then press “Shift+Enter”



The terminal might look different.

4)The script will now install all the dependencies needed,

5) The packages will then be tested by being loaded in



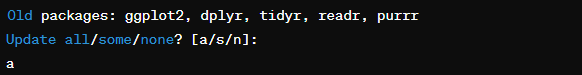
6) It will print “done” when the script is finished



## Common problems

A package update could return an error, most of the time this is due to one of the following problems:

* There needs to be more storage space for the update.
  + Try clearing up some space and try again.
* There is an update prompt asking if you want to update all/some/none of the packages.

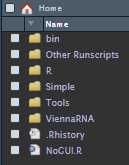


* + Try to respond with all if it doesn't work. try to update the packages by hand using the command: update.packages(“name of package”)

### Can't open plfold\_dp.ps for dot plot

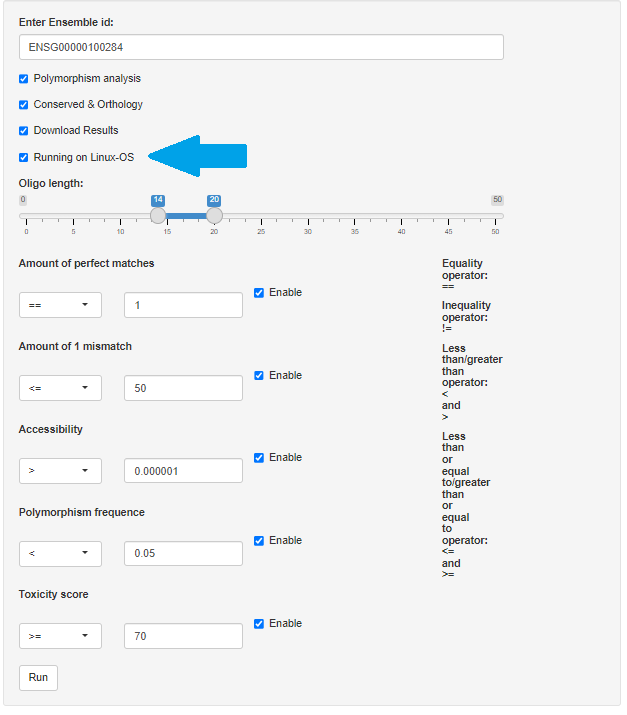
If this is the case, most of the time this is due to storage shortage or due to the script being runned in the wrong directory. This might be some of the solutions that have worked before:

* restarting the server.
* try clearing up some space.
* change the work directory to the folder where viennaRNA is: setwd("/home/gebruiker")



# Running on a non-Linux operating system.

When running the script on a non-linux operating system (windows or mac) the Secundary structure prediction tools wont work due to system commandos specifically for linux. running the script is still possible by turning this feature off, this can only be done in the GUI version by toggling the ‘Running on Linux-os” box on or off.

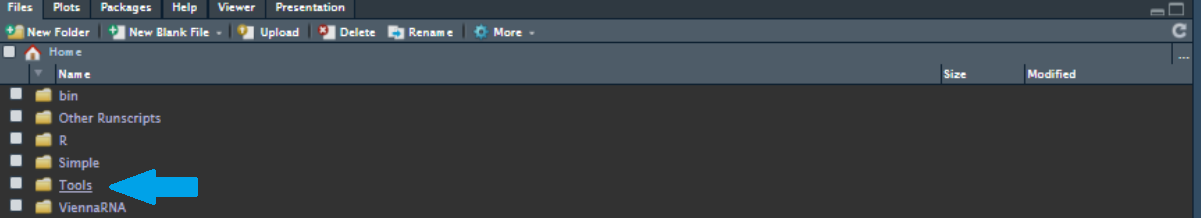


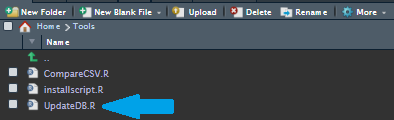
# How to update the databases

## Updating the databases can be a little more difficult.

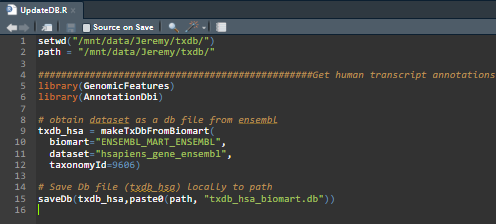
To update the DB file you can use the “UpdateDB.R” script in the Tools folder.  
**Make sure to backup the old db file before removing it from the original map.**simply run the script as you would do as the “installscript.R”

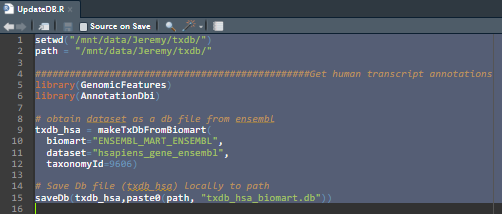
1) Simply open the script found in the Tools folder in the lower right.



  
2)When the script is opened select all the lines using the button combination: “Ctrl+A” on

Windows/linux and “Commmand-A” on Mac.





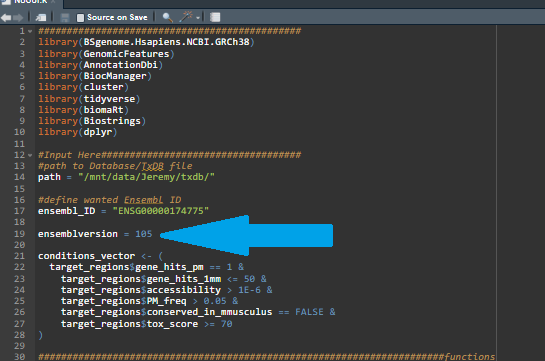
3)Then press “Shift+Enter”, the script is finished when “done” is printed in the console.



4)Make sure everything still works.

## Updating the Ensembl version

**is not advised without doing deep research, some versions are missing crucial information** for the tool. A stable version is 105.  
To change this, go into the script and search for the following line:

**

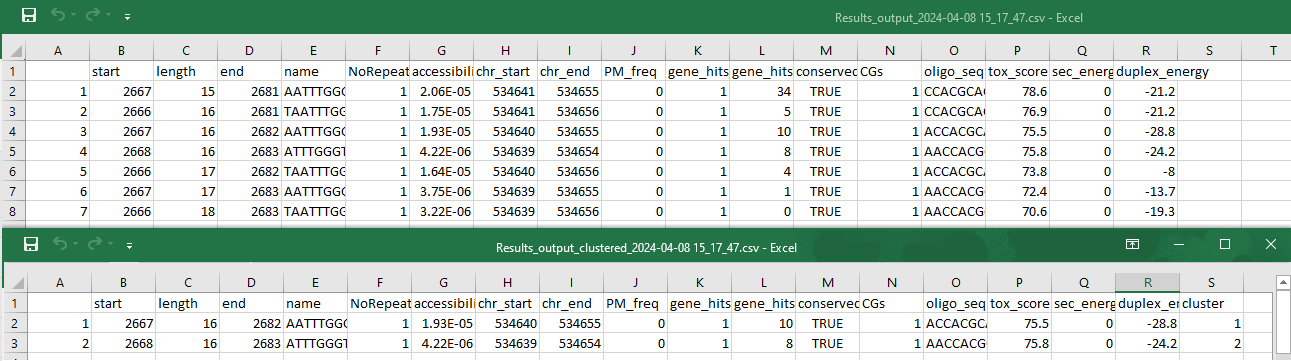
Here you can change the version to whatever is preferred.

As of April 5th, 2024, only version 105 has been tested for a stable version.

Other versions may result in an incorrect working script.

# 

# Results example



# FAQ

**Q1: What do I need to start using the system?**

A1: You need access to a computer with an Erasmus MC internet connection to connect to the Linux server.

**Q2: How do I connect to the Linux server?**

A2: Use your SSH client with the server's IP address and your credentials. Detailed steps can be found in the "How to use the tool" section of the user guide.

**Q3: What is the ASOs Design Tool?**

A3: The ASOs Design Tool is a custom R script available within the R Studio Server environment. It helps find areas in genes where antisense oligonucleotides (ASOs) can be used, offering suggestions for possible ASOs to assist in research.

**Q4: How do I update the system or install new tools?**

A4: Updates and installations are managed through the "Installation Script." Refer to the "How to update the databases" & “How do I install the software and its dependencies” sections for step-by-step instructions.

**Q5: Do I need any specific software to use R Studio Server?**

A5: Yes, the R Studio Server requires the R Software Environment to function. This environment is pre-installed on the Linux server.

**Q6: What is ViennaRNA and how do I use it?**

A6: ViennaRNA is an external tool integrated into our system for RNA structure prediction. It's accessible from the R Studio Server for conducting various analyses.

Troubleshooting ViennaRNA is best done by contacting the developer's email:

“rna@tbi.univie.ac.at”

**Q7: Can I access the system from any PC?**

A7: Yes, as long as you’re connected to the Erasmus MC internet, have SSH access, and the necessary credentials, you can connect to the Linux server from any PC.