University of Torino

UNIVERSITÀ
DEGLI STUDI
DI TORINO

Doctoral Thesis

Reproducible Bioinformatics Project

Author:
Neha Kulkarni

Supervisor:
Prof. Raffaele A. Calogero

A thesis submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy

in

Complex System of Life Sciences

December 2019

“*You started this very well…end it gracefully!*”

~ My parents – Sujata and Satish Kulkarni ~

Declaration

I hereby declare that this thesis has been fully composed by myself and it has not been submitted for any other degree or professional qualification. The work described has been performed by myself, except where expressly indicated otherwise, and all sources of information have been appropriately acknowledged and references provided.

Neha Kulkarni

NKulkarni

Acknowledgements

The work in this thesis would not have been possible without the scholarship awarded by the University of Torino and for their financial support. I am very thankful for the continuous support and guidance of many important individuals who relentlessly contributed to the completion of my research. I am grateful for this opportunity and would hereby like to thank all of them – academics, administrative staff and PhD cohorts in the Molecular Biotechnology Centre and the department of Complex System of Life sciences.

First and foremost, I am deeply grateful to my supervisors, Professor Raffaele Calogero for his guidance, insightful comments, motivational advice and continuous support throughout the entire process of this PhD. His close supervision helped me complete this journey successfully. The joy and enthusiasm he has for his research was contagious and always motivated me, even during tough times along my PhD pursuit.

I would also like to take this opportunity for thanking Maddalena Arigoni for always being near and a dear, sisterly figure, constantly caring at all times. With her bright intellect in Biology, she was a continuous inspiration to both my fellow lab mates and myself. She has contributed immensely to my personal and professional evolution at MBC Torino.

I would also like to thank Martina Olivero for her scientific advice and knowledge, as well as for the many insightful discussions and suggestions she provided me with when she visited our lab. Her tips on making interesting presentations has proven to be amongst the most useful things I have learned during my PhD.

I would also like to thank Professor Marco Beccuti and Professor Francesca Cordero for sharing their knowledge whenever needed. I have a deep admiration for Professor Francesca Cordero for being one of the smartest and successful women I have ever known.

I would also like to extend my sincere gratitude to Professor Michele De Bortoli, who has always been supportive and continuously motivated me throughout my journey.

I will forever be thankful to all my colleagues and friends for whom I will always have the best words, Riccardo and Luca for stimulating discussions, for the sleepless nights when we worked together, and for all the fun we had in these past years. They were always a catalyst for me in the lab, bringing more energy to the work place.

Last, but not least, I take this opportunity to extend my deepest appreciation to my father, brother, husband, late mother and my best friend Cristina Alexandru for believing in me, providing tremendous support and for building confidence in me.

Table of Contents

Chapter 1

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Abstract

Bioinformatics continues to be more and more integrated in numerous biological frameworks and this emphasize the importance of generating reproducible and reliable results. The current project provides evidence about reproducibility solutions in the area of applied bioinformatics in transcriptomics/genomics, and aims to identify factors influencing reproducibility.

Reproducible Bioinformatic Project (RBP) is a no-profit open source project based on three modules docker containers, docker4seq and 4SeqGUI and was developed for enhancing reproducibility in transcriptomics/genomics workflows. In this study docker containers have been used as a platform to embed different tools, packages and dependencies necessary for the analysis of different sequencing data types. The docker4seq package in RBP comprises of a number of workflows designed and directed for the analysis of mRNA, miRNA and ChIP-seq data sets, confering reproducibility to each individual task. It integrates command line (Bash) and R programming languages with a user-friendly GUI interface, making it accessible for both bioinformaticians, as well as biologists with little-to-no prior skills in computational analysis.

One of the workflows, implemented in the docker4seq package was used to analyse the miRNAs profiles from four biofluids: plasma exosomes, stool, urine and cervical scrapes. Both common and uniquely present miRNAs were identified in each biospecimen and their differential expression analysis helped in classifying and understanding the distribution of these regulatory elements in the different human surrogate tissues. Moreover, uniquely identified miRNAs were cross-validated with previously published studies and were found to be biomarkers for different diseases, such as diabetes, colorectal and pancreatic cancer.

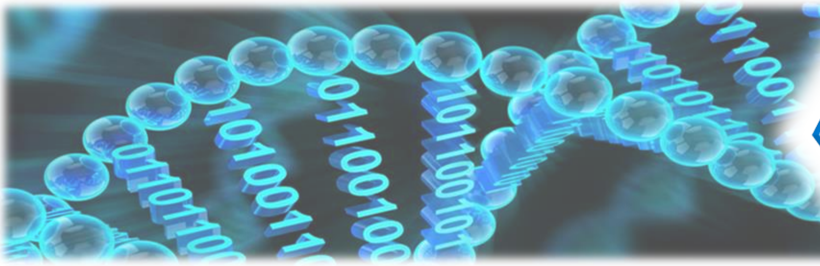
The designed workflows developed and embedded in the docker4seq package were validated and successfully applied in a number of already-published studies, thus demonstrating the versatility, utility, efficiency and scalability of our pipelines. In addition, an increasing number of users reports satisfactory usage of our tools.

Abbreviations

RBP	Reproducible Bioinformatic Project
NGS	Next Generation Sequencing
QC	Quality Control
GSE	Gene Set Gene Enrichment
CERNO	Coincident Extreme Ranks in Numerical Observations
API	Application Program Interface
OS	Operating Systems
SQL	Structured Query Language
VM	Virtual Machine
GUI	Graphical User Interface
3Rs	Reproducibility Replicability and Reapeatibility
NIH	National Institute of Health
GNU	Unix-like operating system
URL	Uniform Resource Locator
KVM	Kernel-based Virtual Machine
GATK	Genome Analysis ToolKit
DRMAA	Distributed Resource Management Application API
RNA	Ribonucleic acid
ChIP	chromatin immunoprecipitation
RAM	Random access memory
DNA	Deoxyribonucleic acid
BCL	Base Call
SE	Single End
PE	Paired End
STAR	Spliced Transcripts Alignment to a Reference
MMPs	Maximal Mappable Prefixes
SA	Suffix Array

SAM	Sequence Alignment Map
BAM	Binary Alignment Map
BWA	Burrow Wheels Aligner
SHRiMP	SHort Read Mapping Packag
SWA	Smith-Waterman Algorithm
cDNA	Complementary DNA
RSEM	RNA-Seq by Expectation Maximization
HUGO	Human Genome Organisation
DEseq	Differential Expression Sequencing
IgG	Immunoglobulin G
FDR	False Discovery Rate
MACS	Model-based Analysis of ChIP-seq
hsa	Homo Sapiens
mmu	Mus Musculus
PCA	Principle Component Analysis
ALS	Amyotrophic Lateral Sclerosis
PBMCs	Peripheral Blood Mononuclear Cells
ncRNAs	Non Coding RNAs
NTHi	Nontypeable Haemophilus influenzae
BC	Bladder cancer
DE	Differential Expression
UTRs	Untranslated Regions
PDEx	Plasma-Derived Exosomes
STEM	Specimen Transport Medium
HPV	Human papillomavirus
GRanges	Genomic Ranges

Chapter 1



Introduction

Chapter 1

Introduction

The term Reproducibility reminds me of a very famous story of the 17th century, when famous Irish chemist, Robert Boyle was working on “vacuum” considered at that time to be a questionable topic. Although two other well-known scientists – René Descartes and Thomas Hobbes believed that vacuum did not exist, Boyle designed an air pump to generate and study the vacuum, and continued to repeat the same experiments time and time again, believing that others will eventually trust them as facts. It was during a time like that, when building an air pump was overall cost inefficient and difficult, that the first documentation of “reproducibility” was recorded. Similarly, in 1660s the Dutch scientist Christiaan Huygens from Amsterdam build his own air pump, that being the first time an air pump was built without Boyle’s direct support. Huygens’s air pump seemed to be a success, as he performed experiments that defined the effect of "anomalous suspension of water", through which he observed that water in the glass jar inside the air pump seemed to drift away.

During the same time in England, Boyle failed to replicate the same experiment with his own-designed pumps, thus making Huygens’s observations on “anomalous suspension” hard to accept by the scientific community. Hence, Huygens was invited to perform his experiment in England in 1935 and with his collaboration and expert guidance, Boyle was able to replicate the anomalous suspension of water (Schaffer S. *et al.*, 1985). With all these scientific allegations, it expressed the need for reproducibility as the necessary condition for demonstrating a scientific fact. At that time, a famous statistician Ronald Fisher for the first time wrote the book ‘The Design of Experiments’ in reference to reproducibility for statistical scientific. (Fisher R., 1971; Popper K., 1992)

In Science newsletter article of May 2015 – ‘What does “reproducibility” mean?’ reproducibility was defined as the ability to reproduce an experiment for achieving exactly the same results, independently of the location and the operator. In simple terms, users could try to replicate an analysis using the same raw data and same pipeline(s), in order to produce same results and confirm the reliability of the method.

These aspects always remind me of situations in which I have tried to reproduce my own findings and of the importance of obtaining comparable results, supporting my hypotheses. For example, my colleagues and I have faced difficulties in replicating differential expression analysis results for RNA-seq data, performed a few months before. To our surprise, we were unable to reproduce our initial findings using the same code, the same data, the exact same tools and computer. We checked file by file and every line in the code, trying to understand what were the reasons behind the variation in our results on the same data sets previously analysed. Eventually, we realized there was one thing in particular we had not paid attention to, and that were some R packages, which had been updated. After reinstalling the old versions we had used before, we could confirm our initial results. This made us realise that the code and data used are not the only important aspects in reproducible studies, but also the software versions and the depending packages. These are pivotal in terms of reproducibility.

A similar incident, named the "Duke Saga" was reported at the University of Duke" by Dr. Anil Potti and his colleagues, who had developed a method for studying the gene expression information from high-throughput microarrays of cancer samples. They had used the data for detecting and anticipating responses to the different drugs used in chemotherapy. At that time, their method was considered revolutionary in the sector of personalized medicines, but it posed problems when other two biostatisticians – Baggerly K. A. and Coombes K. R. (2009) tried to reproduce Duke's studies. They found contradicting results and concluded the initial studies were flawed and did not pay careful attention to data reporting, and paper documentation production. Thus, they reported there were not sufficient proofs to support the findings, which failed to be replicated, based on the following points: (Buffalo V., 2015)

a) The whole genes list was demonstrated to be shifted down in relation to the identifier.
b) Two outliers identified were not on the microarray that was used in the study
c) There was utter disorientation in the treatment and dates recorded for the microarrays
d) The experiment group names were mislabelled.

Keith Baggerly and Kevin Coombes both have very well complied their studies and highlighted that how most common errors are simple, conversely, most simple errors are common in Duke's research article. They also emphasized that poor documentation can not only lead to inconvenience but can be misleading for other readers. In their research article that have provided 5 different case scenarios where the results have simple errors which might put patient life at risk. Its time now we consider "Reproducibility" an important issue, especially in health sciences. (Baggerly K.A. and Coombes K.R., 2009)

The above embarrassing stories highlight why reproducibility is required. Computational Analysis in pre-genomic era, data size was very small, and data were shared only using hard drives, CDs, but nowadays, data size can rise to terabytes. In mid 2000s the Sanger sequencing was slowing getting replaced by less expensive sequencing technologies. After the data size increased and work around the sequencing became a routine, bioinformaticians were involved in developing automated tools and software that could check sample quality, perform statistic analyses, annotate features, build reports and store results in. Before bioinformatics, such work was very laborious and could only be performed for a limited number of genes at a time.

However, it stills seems a difficult task for the biologists to perform the bioinformatics analysis using tools and software as they lack the mathematical and statistical background to understand how a particular package works. Many tools have only command line operations; therefore, tools are chosen based on easy to use. Many biologists use the default parameter that are available with the tools, this might not always be right, the default parameter worked best for the test dataset that the developer has used, as each dataset has its own challenges and limitations, this can also lead to inappropriate results.

Some Biologists might design their own pipeline to analyse the data but they should also consider the fact as to how they want to share the pipeline and update it from time to time, in most of the cases the pipelines are not updated and left alone and often the developed pipelines are not even released, adding to the difficulty level for the reviewers and reads to wonder 'how did they get the results'? (Preeyanon L. *et al.*, 2016).

1.1 Motivation for research

Biological data is considered to be one of the 'big problem', but in reality, it is just one side of a bigger problem. It seems that researchers are having a hard time analysing the big data due to the lack of infrastructure and understanding of the data in such a way that it can be reused. The invention of microarray technology for the first time in biological sciences was producing a huge amount of data, which could be understood and analysed only with sufficient amount of training specifically given in that domain. The advent of NGS produced an explosion of data generation. Analysis and interpretation of such vast sequencing data still remains challenging as they rely heavily on complex computer techniques. This eventually motivates us to take into consideration that researchers should be able to repeat their analysis end to end. (Nekrutenko A. and Taylor J., 2012)

Moralistically, reproducibility is an extremely important term, not only for science, but also as a scientist it makes research more attainable. The basic requirement of reproducible studies is that one should at least be able to reproduce his/her research, this will help to increase the productivity. This will in-turn consistently bring into practice to make it a habit. Lastly, the reviewers could also be given the access to the methods, scripts, software to actually reproduce the results when reviewing the research papers, which will lead to increasing trust, passion towards research and citation of the work. (Sandve G. K., *et al.*, 2013)

Karl Popper once said the best that, if you are the only person who can produce results and others find it hard time producing the same result, then it has not much significance in science. It also causes no trust with such results. It is audacious to build a theory on such an unconfirmed finding, and is a waste of time, efforts and resources to build a hypothesis/theory on such a result. The capabilities of the inventions are in the convenient use of methods/tools for other researcher to administer to their own problems. Science is cohesive, other should be able to pick it up from the piece where one has left to continue to produce new findings, it should be as transparent as possible. (Popper K., 1992)

1.2 Aim and scope for research

Addressing the reproducibility problem is challenging and may require multi-way approach. In this thesis I aim to address the reproducibility issue using docker images and R packages to allow reproducible results in transcriptomics and genomics (Figure 1.1). I also aim to articulate the challenges and help aid the implicit and explicit requirement of reproducibility to support reproducible bioinformatics workflows and guide the community forward in increasing reproducible research in field. Although, my focus here is on the transcriptomics and genomics data processing, principles and challenges also more broadly apply to other omics studies as well. The Reproducible Bioinformatics Project is a non-profit and open source project. I have implemented the workflows in R-functions embedded in a package available at GitHub-repository.

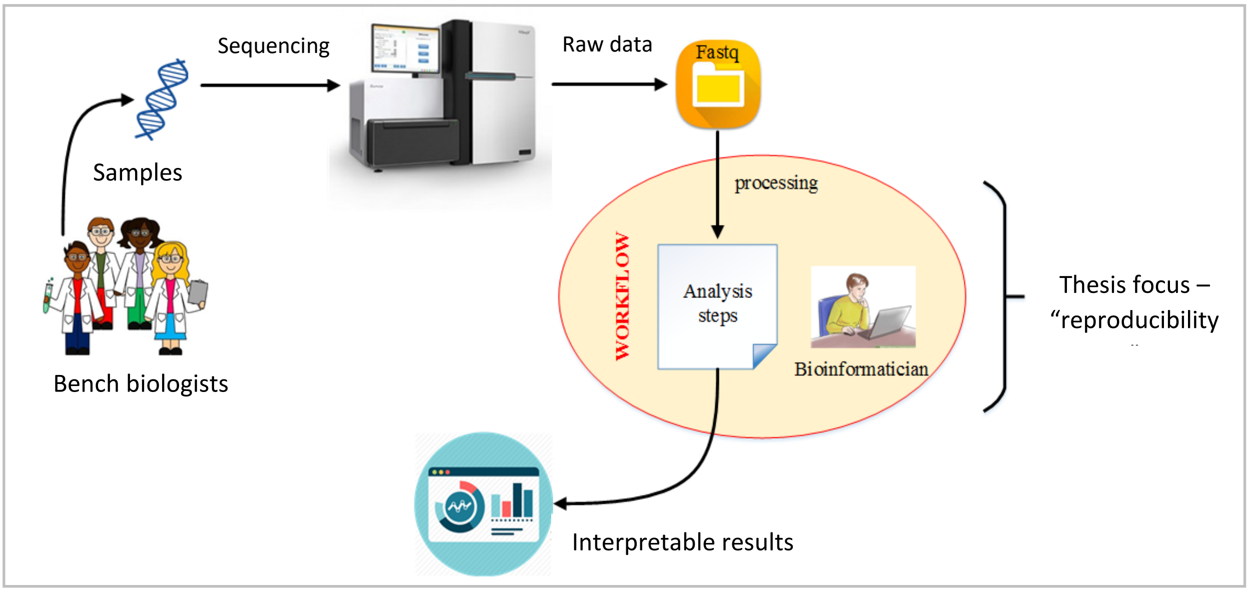


Figure 1.1: A specific reproducible bioinformatics workflow in a biological experiment

Reproducibility is not only useful in biological sciences but is also useful in extended fields. In this project we have tried to answer the questions as to, how we can implement reproducibility in different NGS workflows, highlighting important criteria that make the workflow reproducible-friendly .

The scope of reproducible studies is very vast and is been actively investigated in various fields. The researchers at the National Cancer Institute in France were working on project which provided complete diagnosis and interpretation for example QC or variant knowledge annotation for oncology studies, they wanted to deploy this existing workflow to most of the French hospital laboratories. The computational tool that they used was command-lines and workflow systems in Galaxy, special care was taken in embedding the exact same version of tools and packages. This workflow embeds large number of tools and make use of many databases at the same time. The change in the database version implies an update, development and validation of a new stable version of the workflow. A regular maintenance of the workflow is done and ensured with a test dataset to validate the tools. (Cohen-Boulakia S. *et al.*, 2017)

A similar study was conducted on the same lines of reproducibility by a different group researcher. The team evaluated eight different algorithms for analysing Gene Set Gene Enrichment. Out of the 8 they identify CERNO a novel algorithm which outperformed the rest of the algorithm tested along with it. Coincident Extreme Ranks in Numerical Observations (CERNO), was not only considered to be highly reproducible but was also sensitive and fast. The comparison study conducted by the team was concluded by ranking CERNO highest in terms of reproducibility (Zyla J. et al., 2019). Such studies have helped me in designing my project.

1.3 Significance of study

Essentially science is expected to be reproducible, but it is not tested all the time. Every new finding is the base of the existing discovery. The published data and literature act as stepping stone for the new discoveries. Researchers consider the published literature as gold standard to build their new hypothesis. When the published data is some much relied on, it is highly essential for it to be validated and utmost accurately reproducible. In short, reproducibility is very essential, so that none of the researcher waste time is reproducing any experiment which is not validated before and provide enough evidence of the experiment and result being reproducible. (Jalees Rehmann, 2013).

This thesis will provide an essence of the rules for reproducibility of bioinformatics workflows that are widely used in genomic workflows. The primary aim is to provide a baseline to identify a preparatory set of factors that can advance reproducibility in genomic workflows. We intend to present more constructive, non-hypothetical access to reproducible genomic analysis. The effort of us help researcher to make their study end to end as reproducible as possible.

1.4 Problems in study

By now we have a basic idea as to what is reproducibility and how important it is, we also have a grasp of ways to make it work. There is a large number of difficulties that drive reproducibility crisis in biological sciences. These difficulties are not easy to crack, some are environment, data and model availability, pressure to publish, gold standards, and many more. Each of these issues requires a focused solution and, in many cases, there is no single rule solution to the problem. (Popper K., 1992). Although it seems that computational reproducibility seems to be an easier aim compared to applying reproducibility in experimental biology, the increasing complexity of softwares brings challenges in reproducibility also in computational biology (Boettiger C., 2015).

1.5 Ten simple rules for Reproducible Computational Research

Our goal as bioinformaticians is to be of help to biologists, who have little or no knowledge about computational biology. For such biologists, to have a pleasant experience with computational tools, *Sandve GK* has laid down specific rules to structure command line operations that will be useful for biologist and which are useful to make it reproducible. Now presenting in detail ten simple but most important rules to work with while trying to make your study reproducible. (Sandve G. K. *et al.*, 2013)

1) Results should be trackable

Making a note of the steps involved in performing experiments producing promising results, is always the best to begin for reproducible studies. These steps might involve many interdependent steps starting from collection of raw data to the end result, for example: in bioinformatics studies, one-line commands, end to end scripts and finally the program. Therefore, all the steps whether performed as workflow or steps involved in pipeline or even manually, need to be recorded.

2) Avoid manipulating steps manually that are involved in analysis

The most efficient way to be able to reproduce results is to avoid as much as possible any step that involves manual writing steps.

3) Record the exact version of the tools used

To be able to reproduce the results exactly, one need to know or record the exact version of the tools used in the original work.

4) All the scripts should have a version

When the codes are in the development state they need to be versioned in an appropriate fashion. In short, every piece of code needs to be named and stored as to when it was developed and for what reason.

5) Recording the results in between in a standard format

Theoretically, being able to fully trace back the code, the mid-way results can be anyway re-established. However, storing the results and skimming through the results can help understanding what exactly have we done in the script to achieve a particular result.

6) Note and provide random seeds

When random numbers generation is required the initial seed should be stored, so that all the random numbers generated in the analysis could be reproduced.

7) Store raw data behind plots

Plots from time and now have been very useful in the reading data at a glance. They provide an overall understanding of what the data is. The plots are also considered to be a major tool to for visualization, and most of the time they are modified to improve visual readability of the data. If one stores the raw data behind the plots, it is easier to have access to original data, further one can just make changes in the script to allow different visualization on the data that is processed, where the original data still remains intact.

8) Generate a hierarchical analysis output, allowing transparency at every step

A step-wise article can be made for all the data, plots and results that are involves on the study. This can be easily achieved by simple adding an html file along with a hyperlink at fig, data, result to provide a detailed description of the same. This creates a well-versed transparency of the analysis performed.

9) Providing the textual statements to the results

Usually, the result of the analysis performed, and the respective interpretation are disconnected, the results are placed always along with the data and the interpretation in the text format is not there usually along with the results, but could be present in the form of personal notes somewhere unrelated. Textual interpretations added to the scripts helps understanding the rational of the script even after long time.

10) User access to script and results

Finally, sharing the data results, intermediate results and scripts are what most of the journals now a days allow. It's become more and more easy, accessible and standardized process to allow sharing. Main codes and source codes as supplementary materials. (Sandve G. K. *et al.*, 2013)

Making efforts using the above rules, will only make the reproducibility process easy and accessible. Moreover, it provides a strong essence of the study being exceptionally true, good quality and transparent, leading to good journal that other researchers will cite study.

1.6 Insightful Introduction to Docker

There goes a very preliminary thought on what is docker, it’s a blessing for today’s complex systems. It provides excellent facilities, that are useful for developers. It poses the ability to build, distribute, run and is a compact lightweight packing tool. It is relatively cheap and now a day’s researchers prefer docker containers over traditional virtual machines. Docker is called the ‘hotter than the hot’ as makes desirable for more and more apps to run on the same old server and is believed that it goes easy on packaging and distributing. (Vaughan-Nichols S.J., 2018)

In June 2014, the first version of Docker was released. Public and privat institutions are moving from traditional virtual machines to Docker at an expedite rate. According to recent reports, over 3.5 million applications have been placed in containers using docker technology and about 37 billion containerized application have been downloaded. There are more than 400 research papers, which have proven the utility of docker containerization.

Container history starts back in 2000s, when Oracle Solaris had similar technology called ‘Zones’ while the other companies like Parallels, Google, and Docker worked on projects like Linus containers develop containers to more efficiently and robustly. With container system, one can have four to six times the number of server application instances on the same machine. Secondly, containers are easy to pack, deploy and run application as a lightweight, independent application, which can practically virtually run anywhere. The prime difference in the containers and VMs is that VMs is managed by a hypervisor abstract machine, while the containers provide an abstract OS. (Figure 1.2) There is one thing that VMs can do but container cannot. VMs can practically run the different OS, in dockers all the containers use the same OS and kernel. (Babak B. R. *et al.*, 2017; Vaughan-Nichols S. J., 2018)

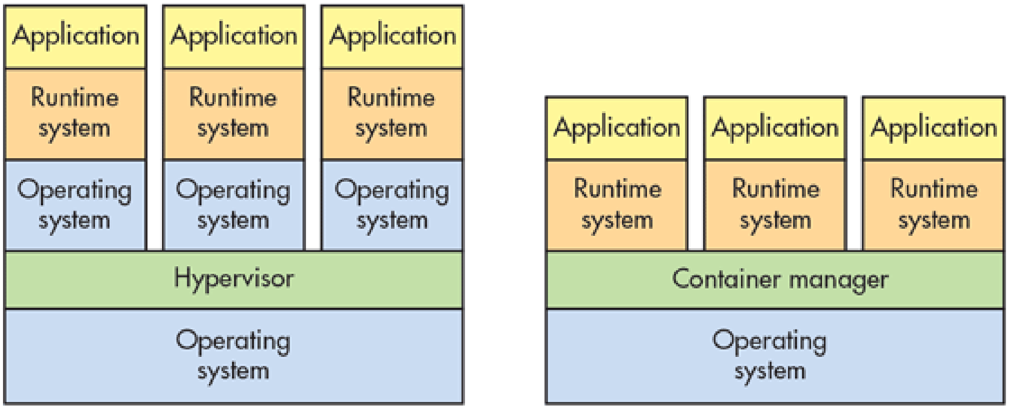


Figure 1.2: VM and docker component

Docker Architecture

There are four main components to the Docker, Docker Client and Server, Docker Images, Docker Registries, and Docker Containers. The different components of the Docker work in harmony to provide full functionality to the Docker in whole.

Docker Client and Server

The client is the terminal interface that provide the command to the docker server to process the tasks accordingly. As depicted in the figure below, the client can be multiple and can be on the same machine as the server is or can be remotely connected to the server which is running on the other machine. The main task of the docker clients is to pull the images from a registry and allow it to run on the docker host (Figure 1.3).

Some of the common commands of clients are as below:

```
docker build
docker pull
docker run
```

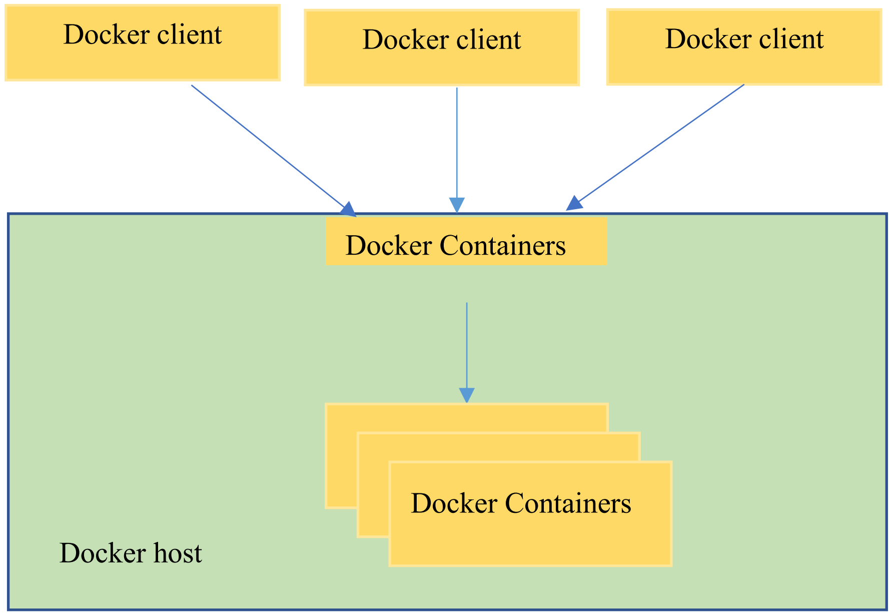


Figure 1.3: Schematic representation of Docker client, containers and host

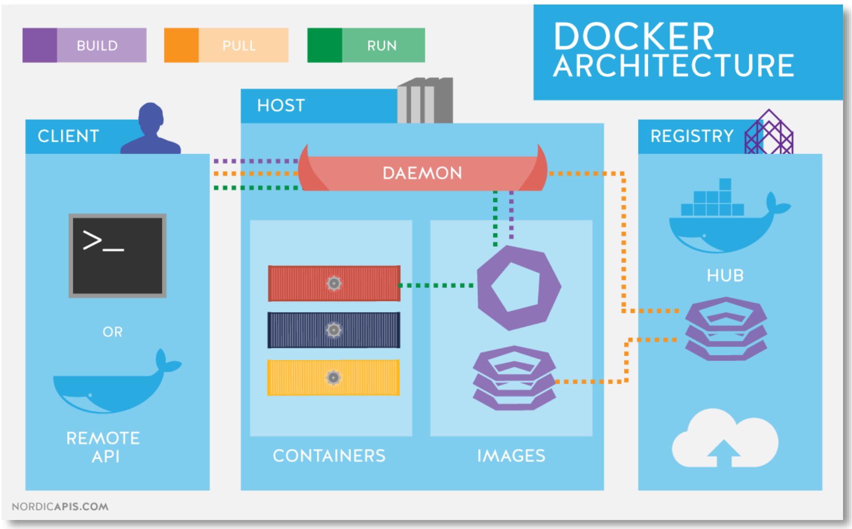


Figure 1.4: Docker Architecture

Docker Host

This is the place where all the action happens. The Docker Host provide a suitable and defined environment to carry out task and run the application. It consists of docker daemon, images, containers, networks and storage. The daemon carries out all the container-related action and obtains commands from client and the remote API. The daemon uses the above-mentioned commands to pull and build the container images as appealed by the client, daemon creates a running model for container with the commands obtained called a build file (Fig. 1.4).

Docker Objects:

a) Docker Image

A docker image is a file having multiple layers, it is used to execute the code from the docker container. There are two ways of creating a docker image:

- Create a Dockerfile
- Run some type of build command that uses the Dockerfile

So basically, docker file is a recipe for creating the docker image and the docker image created by running the command that uses the dockerfile.txt and lastly, docker container is the running example of docker image. A new image can be built and the necessary applications can be added to the base image, process of generation of new image is known as "committing", if you do not committee the change the changes are not recorded.

b) Docker Registries

The docker registries consists of docker images and acts like a register or versioning of these images, facilitating their storage and distribution. The images are pulled and pushed from and to registries. The two types of registries are public and private are available the Docker hub which is the public registry is used default when installing the docker engine, users can pull, push images without generated their own images from scratch. The Docker hub is the feature which helps the images to distribute to public or private areas. (Babak B. R. *et al.*, 2017)

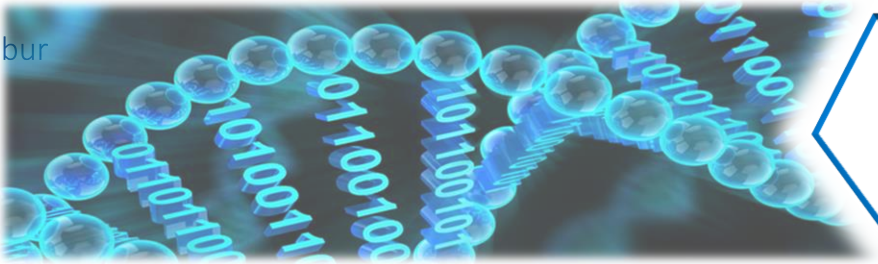
c) Docker Containers

Docker containers are boxes that the docker images creates. These boxes are meant to hold the entire applications, so that these applications can run in a confined way. For eg: one is working of the image of the Ubuntu OS with SQL server, when the ubuntu image is used to run using the 'docker run' command, a new container is created. The containers have a limited access to the stored resources inside them, but the access can be extended by building an image into the container. As docker containers are smaller than VMs, it is easy to set it go in a moment and the outcome is better server density. (Fig 1.5) (Babak B. R. *et al.*, 2017)

1.7 Thoughts on Docker and Bioinformatics

BioDocker project aim to create many docker containers consisting of bioinformatics tools ready to be distributed to maximise reproducibility in the field. The main of the project is to expand the use of docker containers in the field of computational biology and bioinformatics and to homogenize the bioinformatics tools. The project came into existence in 2014 by ‘Felipe da Veiga Leprevost’. Similarly, another project BioBoxes was developed to make the bioinformatics studies reproducible and institutionalize the software packaging methods. What BioBox says is that analysis of biological data using the bioinformatics tools has made the entire bioinformatics to be so much reliable on the software is that there seems no way we can avoid the thought of reproducibility, therefore the problems associated with the software have come in handy when one uses the bioinformatics software. Problems such as partial codes, software installing dependencies and irreproducible workflows, all of these issues provide user an extremely bad experience. Therefore, Bioboxes considers software containers to have a high potential in solving these problems. (Keith Bradnam, 2015)

Chapter 2



Literature Review

Chapter 2

Literature Review

By now we know the subject matter of reproducibility and its importance. Although when shed some light upon the case to case-based scenarios in different journals help one to understand what else can be done to make the reproducible studies impactful. The literature survey helps to set certain rules around the problems, it also highlights the problems that was never ever occurred for any other researchers. It provides researchers to be elucidate in the subject. It defines the problem not only in our studies but also in various domains indicating the alarming need for solutions. Lastly, it helps the research to provide quality research. However, reproducibility still remains a big problem to achieve in day to day life and also remains a problem to follow the rules specified by scientist with ease. Digging deep and to dissect certain terminologies similar to 'reproducibility' could be somewhat comforting.

In 1990, Jon Claerbout one of the researchers at Standford, wanted is PhD student to perform experiments that could fulfil the standards of reproducibility. He basically wanted to make sure that this reproducible research be documented in the form of thesis, for the other colleagues to be able to perform reproducible analysis using the same command and the same data and were aiming for a publication. They used 'make' a tool which uses source code and read list of commands from the makeFile to build a software. They build a workflow with the commands burn, build, view and clean and also attached the publication which included programs, scripts, parameter files and makefiles. Clearly the author has made efforts in defining the Reproducibility. He has specifically emphasised on the different terminologies and their misconceptions. (Barba L. A., 2018)

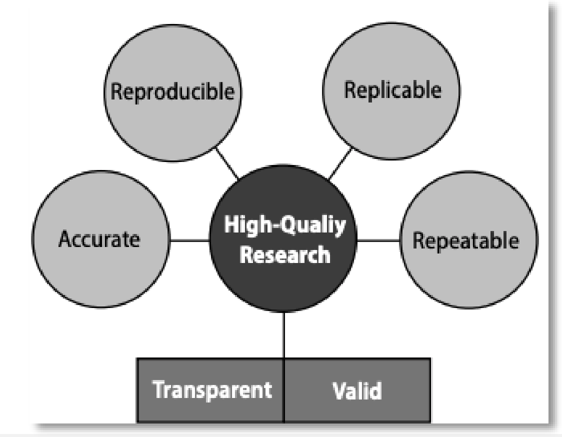


Figure 2.1: Qualities to achieve quality work

The above image depicts the fact that, as researcher we are expected to provide quality of work and this can be achieved only by accuracy, reproducibility, replicability and repeatability (Figure 2.1). (Blatecky A., 2017)

2.1 Reproducibility, Replicability and Repeatability

For a number of years, researcher have highlighted the issue of reproducibility and the confusing terminologies that they often struggle to define. Reproducibility, replicability and repeatability have influenced almost all the fields of science and there was also a survey conducted, where 65% researchers confirmed that they have experienced reproducibility issues in creating an experiment from already published data. (McArthur S. L., 2019) It felt absolutely necessary to define the 3Rs – reproducibility, replicability and repeatability.

Repeatability (same team, same experimental setup): The basic definition for repeatability is that a result can be produced from one experiment given that they are performed with exact same protocol. To illustrate the protocol's repeatability, it requires the following:

- Location: place where the experiment is conducted, a lab setting
- Apparatus: measuring tools
- Techniques
- Observer
- Assumptions
- Timeline as to how long should an experiment run (MacKenzie R. J., 2019)

Repeatability is considered important when one has to compare two methodologies. It is considered that the repeatability of the two methods might have disagreement. For example: if one of the methods have a lot of variation for the measurement for the same subject, it is considered with great confidence that other method in comparison has variation too. Therefore, the two methods do not have a strong agreement. If the two methods in comparison do not have an agreement, the problem is considered critical. An example for this case would be if two methods are developed to measure pulse rate, it would be surreal if they were unrelated.

There is a solution to this problem says the Bland and Altman, to avoid a disagreement in the two different methods, one could take repeated measurement in one method and calculate the mean from them and compare with mean of those achieved from the other method, based on this the two methods can have an agreement or disagreement. (Bland J. M. and Altman D. G., 2010; MacKenzie R. J., 2019)

Reproducibility: The ability to reproduce the results using the same method from one researcher group. This happens in few cases, when one research group has to build a hypothesis on already existing theory, and to do so, they need to reproduce the results that has already be published. If the results are not same, if is considered as mere artefact. Reproducibility and Repeatability and two different concepts in itself. And the two concepts are variably used in various studies.

Replicability (different team, same experimental setup): The results which is obtained by different team, using same protocol, the same measuring apparatus, under the same operating conditions, in the same or a different location. This is quite difficult to attain, although to define replicability is highly important to distinguish between the 3Rs.

Although the two terms replicability and just a smaller part of the larger term 'reproducibility', and they are a part of bigger problem in computational analysis. As expressed these terms need to be defined and identified separately, as they hold completely different meaning in different case scenarios. There, are many such reasons as to why some experiments need to be repeatable or replicable but mostly studies have to be reproducible as they hold great importance for the study to said true and not occurred by chance.

Drummond C. 2009 says that "Reproducibility requires changes; replicability avoids them", which makes absolute sense. Drummond argues that if traditionally seen, the experiments done by different researchers, in different location (lab), using different apparatus. It will certainly be considered that the two experiments are same, but there will definitely have between the experiments due to many reasons. If these differences are removed the studies would be called as replicability, according to him these difference matters a lot in reproducibility studies.

He suggests the researches to look into traditional techniques to clarify the real meaning of reproducibility, unless the true meaning of the reproducibility is not understood one cannot dig deeper to understand features of reproducibility and the process accompanied. Drummond also suggests the issue related to reproducibility can be solved only by discussion in the larger platforms and one of the forums which always these kinds of discussion to take place is Journal of Machine Learning.

Other terminologies

Although, most of the research article rely only on differentiating the key terminologies, but there are other terms which contribute to make the key terminologies to be important. The terms such as accuracy, transparency and validity also hold importance in context of reproducible research.

Accuracy: Generally, accuracy is compatible closely to exactness and truthfulness. It implies precision of the result or data collected for an experiment. In terms of reproducibility, accuracy is the feature of research that is necessary for correctness and exactness of the result obtained. The degree of exactness defines the quality of the results obtained.

Transparency: Transparency has many definitions in term of reproducible research. Transparency is achievable by disseminating the end to end details about the study. This is doable through journals and research articles. In terms of computational research, documenting every step in building the code, used tools, packages etc would thereby lead to transparency. It's the prime duty of researches to clearly display the results/data or source data with easy access for others.

Validity: The term validity is whether a study was able to deliver the results as intended in the assumptions or not. The validity tells the researcher if the study can be trusted or not. (Blatecky A., 2017)

2.2 Attempts to address reproducibility

The reproducibility issue has hampered multiple fields and is continuing to harm on a vast scale. To be able to address this issue in all fields, it is necessary to define the issue very clearly. Researchers have made efforts in defining the problem. They believe that there is no single solution to this issue, therefore there are multiple ways to work towards this issue, which might be beneficial.

The array technology initially when was new to market and was in trend to be used in experiment was dealing with lots of problems. An attempt to solve this problem, many journals in 2001 made it a requirement for data deposition on the website. Although it was compulsory to deposit the data, the agreement still remained an issue. Between 2011 and April 2012, only

2 of 18 papers were able to reproduce the results involving the microarray data, the main reason for this is that the raw data was not available. There were many cases, which revealed that as the raw data was not deposited many studies either had different conclusion or were not reproducible.

It was remarkable that the National foundation and the National Institute of Health (NIH), have made it compulsory to deposit the data and a full disclosure of software if the researcher want to publish, but there still remains the problem of agreement. This was one of the major attempts made by the research journals. (Begley C. G. and Ioannidis J. P., 2015)

There were few other initiatives taken to reduce the reproducibility problems. The Neuroscience Information Frame was generated to help researchers to find the resources to improve the neuroscience research. The researchers have tried to make a note of the details specific to study design, experimenter blinding, randomization of animals, cohorts of sufficient size, inclusion of controls.

Another group Global Biological Sciences Institute aspires to set up an integrated standard that can be practiced across all the research tools. They believe that such a standard set of rules is important for advancing in the biological science fields and can improve the correct identification of terms, but they also think that having such rules consented world-wide would be challenging. For example MDA-MB-435 cell line is named in the many research articles names as a breast cell line, although it is a melanoma cell line. Overall, there are more than 170 articles that have incorrectly defined these, with only 47 correctly recognizing them.

A recent attempt in Reproducibility research was initiated, where the researchers investigated the reproducibility of the cancer Biology publication between 2010-2012. This study is beneficial as they provide the information as to who can perform these replication studies and how they are best performed along with that they have provided a platform for research to conduct debate to clarify doubts within the community. Ten rules laid by Sandve has laid foundation for researchers to follow and make studies reproducible and trustful.

There is a unique approach provided by the academia.edu, they aim to build a domain, where the publications are made available and the peer reviewing is done after they are published. They are also aiming to make all the researches freely available. The NIH seems to be promising reproducible solutions, such as making an integral grant application review, transparent access to data and online platform to discuss the recently published papers and associated problems. The measures take still seem to be in the evolving state, but feels satisfies as they are all embraced by the scientific community (Begley C. G. and Ioannidis J. P., 2015)

2.3 Computational tools and resources supporting reproducibility

Many researchers use various tools and resources to make the studies reproducible and to overcome the issue related to computational reproducibility. The tools and the techniques used could be challenging in itself, which includes a simple technique of providing a written document and slightly an advance technique which includes a virtual environment having an OS and all the software to carry out analysis. There are several ways to make the study reproducible, each carrying its own pros and cons and the limitation to use the technique for a specific dataset.

From a practical point of view, often one single strategy/technique is not enough for the study to make it reproducible, but a combination of different strategies make it work. The aim of addressing the computational tools, techniques and resources for the studies to make reproducible, will be vastly useful for scientists who have limited experience in computational biology and will be more proactive to produce reproducible studies (Piccolo S. R. and Frampton M. B., 2016), as our aim of this thesis also reflects the same.

a) The Narratives

The most helpful approach to make sure that the others can reproducible a computational analysis is to supply an end-to-end description of the process. When the researchers mention a computational analysis in a research article they provide a detailed description of the software used followed by the sequential steps they used. These steps help the other researchers to reproduce the analysis and the results. In most cases, when there is user input required for the software to programme to execute, the written steps available makes possible to do the task. Even in the cases where the open source software is automated to perform certain tasks the, written narratives make the other researcher to understand exactly the analysis the software is executing. (Piccolo S. R. and Frampton M. B., 2016)

Moreover, as mentioned before special focus must be paid to the exact versioning of the software used in perfect order of usage and should also mention the software dependencies. The parameters used must also be mentioned correctly. It is also observed that the computer configuration and OS affect immensely, it is difficult to remember these small details therefore it is important that the scientist make a note of it and document it at the time of the working on the project rather than later. This approach is valid in case the researchers are pure biologist and have no experience with programming language

b) Analysis automatization

Command line operations can be used to perform tasks, instead of using single command and record it individually, one can just the make a note of all the commands step by step, even before that few steps to install and configuring he necessary software. This will not only be helpful to others to reproduce the steps but also to the original researcher to track his own scripts. Therefore, providing the detailed script document is much more beneficially than just the narrative:

```
#install FastQC
wget https://www.bioinformatics.babraham.ac.uk/projects/fastqc/fastqc_v0.11.7.zip

# unzip
unzip fastqc_v0.11.7.zip

# make the tool executable
chmod +x FastQC/fastqc

#download BWA and install
wget https://sourceforge.net/projects/bio-bwa/files/latest/download

#unzip
tar -xvjf bwa-0.7.12.tar.bz2

#installing
cd bwa-0.7.12
make install

#alignment
bwa mem ref.fa read1.fq read2.fq > aln-pe.sam
```

The ‘make’ utility helps to confirm with the dependencies which are already available, it also helps in understanding the necessary dependencies that are very important in carrying out the analysis. ‘Make’ is said to be originally developed for UNIX-based OS, later similar utilities were developed for other OS such as windows. There are few such utilities developed for automatic update of the software mentioned as below:

i) GNU Make is a tool which is designed to manage the executables the source program files. Make know to perform job, by following instructions from the file called the makefile.
ii) Snakemake is also a better and an extended version of GNU make, it provides a better flexibility and provides parallel tasks running. They mostly are python scripts.
iii) Bpipe is a tool that controls and executes all the bioinformatics pipelines. It makes parallelism easier and is known to work better with shell scripting.

It is recommended that the scripts be provided along the supplementary information, moreover it is feasible that the scripts be made public and added to public repository and a simple URL permanent link to the repository be provide in the research article. It is often a good practice to add the scripts as version control to the repository and share using the services like GitHub.com or Bigbucket.org. It is not only useful for the other researchers to see how the code or the script is developed throughout the research, but it is often beneficial for the developer to track his progress and get back to the errors easily. For the other researchers it is beneficial in terms of using the bits of version control scripts to develop their own. For the purpose of citation in the final stages the other researchers can simply ‘tag’ the part of the script they have used.

c) Software frameworks

It is very common that all the scripts majorly depend of the external software and OS specifications. In case where the user wants to perform any non-parametric statistical test, it feels very convenient to pull in any library that can do the job, rather than writing the code to perform the non-parametric test. This is easily possible if the libraries are available. It is very important that a specific version of the library is found to reproduce the analysis. Building pre-exiting software frameworks for libraries that are commonly used to perform analysis, can be one way to address the issue. For R language, Bioconductor framework make it possible to have a control over the versions and code distribution. It has hundreds of libraries useful for performing analysis. The library once developed can be integrated with the Bioconductor’s framework and can be used by many other researchers to perform their analysis. The best part of the Bioconductor framework is that it downloads and install the necessary dependencies required by the library. Apache Ivy and Puppet are other examples of software framework.

d) Literate programming

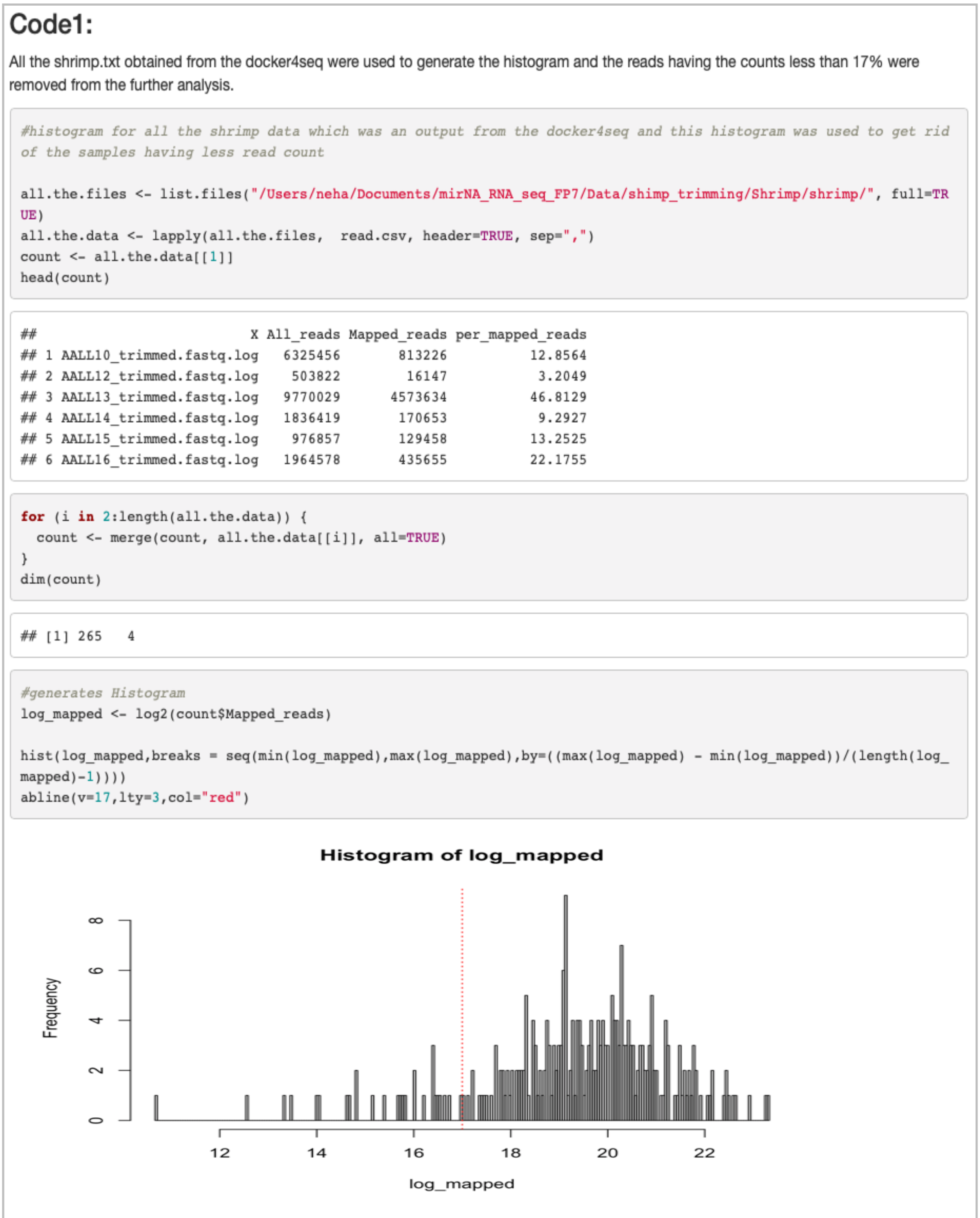
Even if the written document is provided with the code in the research papers, sometimes it is very confusing as to find out how certain section of the code can give an output. Sometimes, even the written document alongside the code seems not useful as the code evolves at many steps. One possible solution for this problem is literate programming, it a method in which the researchers write the narrative and blend the code along with the narrative, once the code is complicated and upon its execution an output files, tables, figures are generated. This helps the other researchers to understand the code and narrative for it very well.

Knitr, is considered very useful and is commonly used among researcher. Knitr is written is R programming language. The knitr functions is carried out in Rstudio which is an R interface which manages the R environment and packages. Knitr generates the dynamic reports which includes, chunks of codes, narratives, plots etc, the reports are often generated in a web-based html and pdf format. (Figure 2.2)

Knitr can be used for simple codes with minimum amount time spent of the code generation. It provides an interactive figures and sufficient amount of narrative. For programs requiring extensive OS environment and advance scripting, tools such as Dexy.it is very useful it helps the code to speak for itself. It is very powerful and flexible. Dexy.it shows of the codes and highlights beautifully the syntax that is used in certain section.

Table 2.1: Other examples of literate programming (Chirigati F. and Freire J., 2017)

Tools	URL
Ipython Notebook	https://ipython.org/notebook.html
Janiform	https://github.com/uds-datalab/PDBF
Sweave	https://stat.ethz.ch/R-manual/R-devel/library/utils/doc/Sweave.pdf



Code1:

All the shrimp.txt obtained from the docker4seq were used to generate the histogram and the reads having the counts less than 17% were removed from the further analysis.

```
#histogram for all the shrimp data which was an output from the docker4seq and this histogram was used to get rid
of the samples having less read count

all.the.files <- list.files("/Users/neha/Documents/mirNA_RNA_seq_FP7/Data/shimp_trimming/Shrimp/shrimp/", full=TR
UE)
all.the.data <- lapply(all.the.files,  read.csv, header=TRUE, sep=",")
count <- all.the.data[[1]]
head(count)
```

```
##                           X All_reads Mapped_reads per_mapped_reads
## 1 AALL10_trimmed.fastq.log   6325456       813226          12.8564
## 2 AALL12_trimmed.fastq.log    503822        16147           3.2049
## 3 AALL13_trimmed.fastq.log   9770029      4573634          46.8129
## 4 AALL14_trimmed.fastq.log   1836419       170653           9.2927
## 5 AALL15_trimmed.fastq.log    976857       129458          13.2525
## 6 AALL16_trimmed.fastq.log   1964578       435655          22.1755
```

```
for (i in 2:length(all.the.data)) {
  count <- merge(count, all.the.data[[i]], all=TRUE)
}
dim(count)
```

```
## [1] 265   4
```

```
#generates Histogram
log_mapped <- log2(count$Mapped_reads)

hist(log_mapped,breaks = seq(min(log_mapped),max(log_mapped),by=((max(log_mapped) - min(log_mapped))/(length(log_
mapped)-1))))
abline(v=17,lty=3,col="red")
```

Figure 2.2: Knitr output example

e) Virtual Machines

All the platforms and tool mentioned above require and are dependent on heavy software dependencies to identify, download, install and configure before any analysis is carried out. This is time-consuming process can be very frustrating for scientists. A whole lot of energy is wasted in configuring the dependencies if the OS is completely different and previously installed incompatible software dependencies are a huge problem in itself. One solution to such problems in the Virtual machines. Virtual machine acts as a simple computer in itself with an OS encased in it.

The VMs are a box with all the software dependencies packed and ready to be executed on any computer regardless of any OS. In certain case where the researcher's computer (host) is an windows machine and wants to perform analysis that is embedded in the Linux VMs (guest), the researchers can easily have control over the (guest) OS and therefore can perform any modification required. After the analysis is performed the researchers can export the entire VM as a single binary file, following to which the other researchers can use the same file to construct exactly same VM environment to reproduce the original results. This system is beneficially in following ways:

a) It is one-time job for researchers to provide a narrative of installation for one OS
b) The other researchers need to only install the Virtual software and nothing else
c) The analysis can be executed multiple times with same Virtual machine

One disadvantage that makes researchers think twice about using the Virtual Machines for reproducibility is that the virtual machine files are high in size mostly gigabytes, in cases where they also comprise of the raw data files. This makes it difficult to share with the research community. One of the solutions to this problem is that making use of cloud computing, where the user can perform the analysis in the cloud repositories, store the VM files and share with other users from the same environment (Table 2.2).

However, some researchers do not feel safe when it comes placing the data on the cloud at least at the time when they are analysing it, they would rather prefer keeping the data on the local machine. The researchers might need to pay fee to activate the cloud-based services. VMs can be considered good options for reproducibility, as they allow to re-execute the analysis along with investigating the scripts, codes and contents present.

Table 2.2: Examples of cloud-based services

Cloud based services	URLs
Google cloud Platform	https://cloud.google.com/
Rackspace.com	https://www.rackspace.com/en-gb/cloud
Amazon Cloud Services	https://aws.amazon.com/what-is-aws/
Windows Azure	https://azure.microsoft.com/en-gb/services/cloud-services/
CloudBioLinux	http://cloudbiolinux.org/

f) Software containers

Software containers are very similar as the virtual machines, they are comprised as a container encapsulating the OS elements, scripts, codes and data ready to share with other users. There is not much difference in the way the VMs and the containers work to produce the outputs. One of the major differences in terms of the containers is that a container provides an abstract OS unlike the VM where it provides an abstract machine which uses the drives targeting the abstract machine. A container needs a latent OS that supports the applications using the virtual memory whereas the VMs have their own OS using the hardware VM supported by the hypervisor. In terms of flexibility VMs are considered better as the containers are designed specific to the OS, although the containers require very less computational process as compared to VMs and they can be initialized much more quickly than VMs. (Figure 2.3)

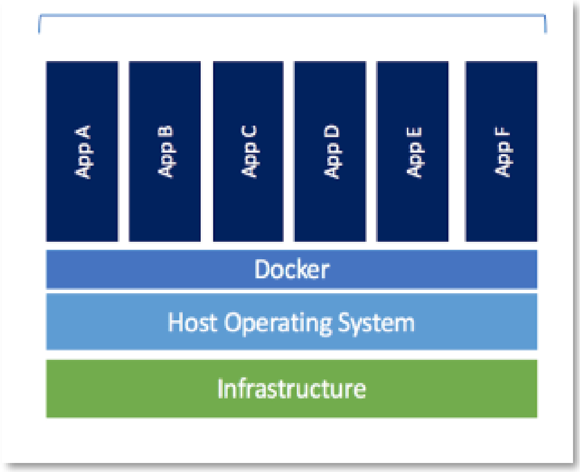


Figure 2.3: Containerized applications

Dockers have gained its popularity since the time they are developed as they have the capacity of providing the content of the user specific dockerfile which is basically text-based file, that can be used by the other researchers in reconstructing the container. And this text based dockerfile can be easily shared, tracked and versioned in repositories. Finally, when the docker container is build the content can be exported in the form of binary file, as these files are very much smaller and can be shared via hub.Docker.com. (Table 2.3)

Table 2.3: Online sources of software containers

Open source container software	URLs
Docker.com	https://www.docker.com/
LinuxContainers.org	https://linuxcontainers.org/
lmctfy	https://opensource.google.com/projects/lmctfy
OpenVZ.org	https://openvz.org/
BlockBridge	http://www.blockbridge.com/

As emphasised before, the use of docker containers comes in handy with some problems. With respect to one single container that is stored plus executed individually by other containers on same computer machine, since all the container are sharing the same OS, the seclusion of the containers are incomplete when compared to virtual machines. What it exactly means is that, a specific container if not given enough access to memory and processing power, then the other containers might have to clash for these resources from the computer machine. This can create more accessibility for containers to face security breaches.

Since Linux platforms are only eligible for the docker containers to be executed, for Mac and Windows they need to be executed within the virtual machines. But if virtual machines are used, there is a possibility of the losing some of the performance benefits of containers. Investment and efforts are going on for the development of refined container technologies. Docker containers seem to have great influence in the near future.

g) Dockers, keep it small and simple (advantages and disadvantages)

It is often debatable as to, ‘why it is important to keep the containers small and simple’? Portability being the prime feature of container, which thereby adds to its popularity. Containers are often compared to the virtual machine containers, in these terms the container always win the battle in terms of preferability, giving rise to misconception that the size of the container does not matter.

Usually, researchers tend to use the default base docker image, which makes the containers large and another main concerns of is that the container eventually duplicates dependencies in application and the image. Also, if the build tools are used within the containers in the running application, this obviously will increase the size of the container and install dependencies which might not be necessary for the application to run. (Ashan Fernando, 2018) There are many advantages of using a small size of the container:

I) For security reasons

In terms of practicality, using smaller containers images are often with a smaller number of libraries inside, this does not directly attack the surface to container. When such a technique is used to build the container, it provides more transparency, allowing one to know what is happening inside the containers. The details about the large size images being vulnerable is seen in the scan log of the container images in the Docker.

II) Efficiency and performance

It is easy and faster to move the smaller containers. Eventually reconstructing the process of build and deployment which will be then pulled to the running container cluster. Smaller containers effectively utilize lesser disk space and memory.

III) Maintainability

With small container image, once the dependencies are installed, it is in full control to be modified at any stages as the configuration modification is known to the application. As there are less libraries installed, it is easy to manage these libraries, keeping them updates along with the OS patches. These are the basic advantages of keeping the images in docker containers small and simple, therefore in this thesis we have implemented the use of small containers images restricted with one workflow in each image. (Ashan Fernando, 2018)

IV) Advantages

In the last few years the need and the progress in using the docker containers has increased. They are high in demand and are popular because of the benefits they provide. Few advantages are listed below:

a) Speed

Containers are mostly appreciated for the speed. It would be unfair not to talk about the speed that containers provide. It requires less time to build a container as they are small and because they are small the development, testing and deployment can be done very fast. Once the containers are built they can be sent to testing. (Babak B. R. *et al.*, 2017)

b) Portability

The applications which are developed insider the docker are easily portable and are portable as a single element. A single container containing it application and dependencies can be all packed together, is independent from the host version of Linux kernel, platform distribution or deployment model. The container can be executed on another machine by transferring it on a machine having Docker installer and up in running, there would not be any compatibility issues. (Babak B. R. *et al.*, 2017)

c) Scalability

The ability of the docker to deploy in many physical servers, data servers and cloud platforms making it very likeable amongst the researchers. It can be exchanged from cloud environment to the local host and back easily and very fast. Modifications can easily be performed. (Babak B. R. *et al.*, 2017)

d) Speedy recovery

The standardized structure of the docker containers, make it possible for the researchers to not worry about each other's tasks. As the responsibility of the programmer is to take care of the application within the container whereas the admin takes care of the deployment and the maintenance of the server with the docker containers. Containers have the ability to work in any environment as they are packaged along with the necessary dependencies for the application, followed by rigorous testing. Predictable results are achieved as the dockers are very reliable and consistent, even though codes can and are moved in between the stages of development, testing and producing system. (Babak B. R. *et al.*, 2017)

e) Density

Dockers are considered to have a higher performance than virtual machines because of their efficient use of resources, the absence of a hypervisor, their capability to run more containers on a single host. (Babak B. R. *et al.*, 2017)

V) Disadvantage

Linux kernel which is the local host of the docker system does not allow a complete visualization. The docker systems requires a specific configured machine, it supports only 64-local machines, they do not support the old machines. A complete visualization can be provided by the docker containers for windows and the machines. There is another tool called boot2docker which can fill the gap for virtualization. (Babak B. R. *et al.*, 2017)

Another feature, which is marked as the feature request which is in progress, that is container self-registration and self-inspecting. Cross platform issue is been an issue ever since, if the application is designed to run on a docker container on windows machine, then it cannot run on a Linux machine and vice-versa. Docker is basically designed to host application, and it uses command line only. The graphical user interface is possible to run within a docker container; however it is vert clunky. Therefore, if an application requires a good GUI docker is bad option. (Babak B. R. *et al.*, 2017)

Docker Performance

There was an experiment performed by researchers where they used two servers with same configuration in a cloud environment and compared the performance of each one of them. One of the servers was used for docker and another one was used for Open Stack platform for KVM a virtual machine tool. The concluded their study by stating that the VM works independently, which makes it easy to apply and is good in terms of network, security etc. Their major highlights were that the docker does not have guest operating system and hence it takes very minimal time to distribute and gather images, the boot time is also very short. Therefore, these being the main reason for docker cloud being favourite for the researchers over the VM cloud. (Babak B. R. *et al.*, 2017)

Another investigation performed by a different group of researchers, wherein they performed a comparison between the Linux containers and Xen virtualization technologies. Xen would be a better choice in terms of equal distribution of resources, it is executed on the same machine and is not depend on the same machine. But, they debate stating that Linux container are ‘n’ times better in terms of utilizing the most of hardware resources and executing the smaller isolated process.

The future is Docker technology. As and how the researchers understand the worth of the docker and its capabilities, they would consider replacing the old virtualization techniques with docker technology. It has many positive features, which has already made it so popular among most scientist. To be able to use the docker efficiently, it is necessary that one makes a move from the default configuration. (Babak B. R. *et al*., 2017)

2.4 Approaches for developed workflows, definition and executions

Many researchers still are frightened about writing scripts and codes. There are many courses available online, which seems helpful. Many scientists prefer the workflow management software that can be executed using the graphical user software. This software allows the user to upload their data and perform analysis using the available tools and scripts with the software. When the scientist has to perform a multi-step analysis they usually provide the output of one tool as input to the other tool and these series of the steps are comprised in the workflow. The development of these pipeline basically follows three rules and differ on the same components: using syntax, configuration and offering command line and/or workbench interface. (Leipzig J., 2017)

2.4.1 Text-based workflow managers

Pipeline managers have been developed for different data types, such as Cpipe, for clinical genomics data. (Figure 2.4) It is based on Bpipe, a pipeline construction framework, which makes execution very simple by making it almost similar to executing it manually. This has become very famous among bioinformaticians, as they do not need to learn any specific programming language to understand or modify syntax in the pipeline. Additional feature of Bpipe include automatic tracking of the commands, logging off from the input and output files, clear the files from failed execution, removing the intermediate results, creating plots/graphs, notification by emails or pop up message in response to failure. (Sadedin S. P. et.al, 2015)

Other workflow managers not based on a graphical environment include Ruffus, Snakemake, Nextflow and CGAT (core and pipelines), which use Python decorators and packages for pipeline development and optimization. They can support cloud storage and environment management systems, such as conda, thus making the installation, update and implementation of numerous packages easier. (Köster J. and Rahmann S., 2012; Cribbs A. P. *et al*., 2019)

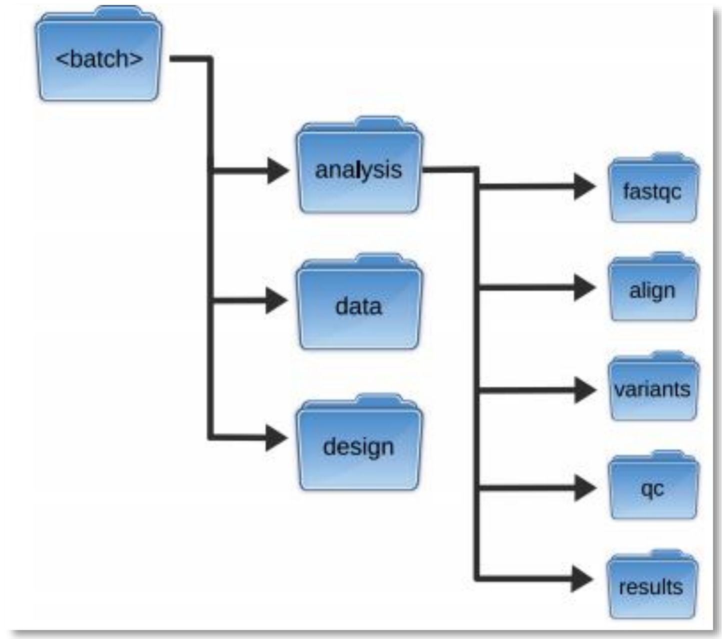


Figure 2.4: Batch directory structure used by Cpipe

Another e.g of the NGS pipeline is the bcbio-nextgen which is an implemented in the python framework. (Figure 2.5) It is a fully automated pipeline which connects with the sequencing machine, run sequences through the pipelines and uploads the data into the Galaxy processing and analysis the data further and for visualization. Another pipeline available is the variant calling pipeline that performs alignment with the reference, identifies the variants using GATK and provides a summary report. Some of the famous project which utilizes this pipeline are CloudBioLinux and CloudMan projects. (Guimera R. V., 2011)

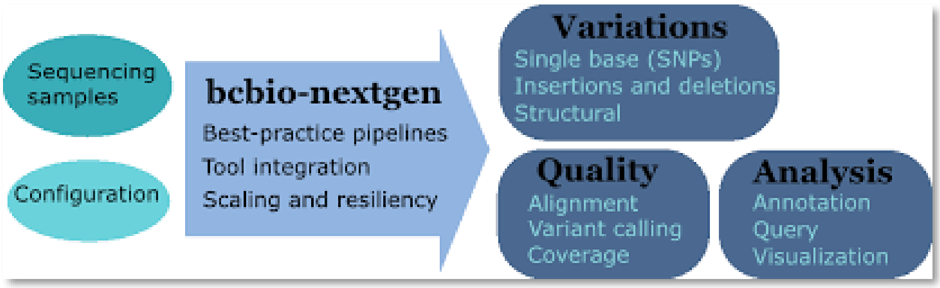


Figure 2.5: Bcbio-nextgen pipeline structure

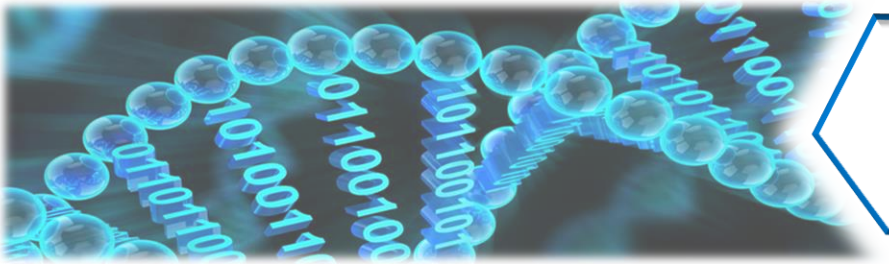
Another very famous python package implementation framework is Omic-Pipe. It assembles scripts in automated, version regulated, parallel pipeline for bioinformatics analysis. The python package it uses is called Ruffus for running the pipeline, Sumatra for version regulation and tracking, Python DRMAA for distributed computing. It is also possible to distribute the python package as a standalone package for installing on the local severs. Omic Pipe, currently supports six published piplines, 2 RNA-seq pipelines, variant calling from whole genome sequencing, WGS from GATK, two ChiP-seq piplines. They also have customized RNA-seq pipelines. (Fisch K. M. *et al*., 2015).

2.4.2 Graphical user interface-based integrative workflows

To easy a bit some challenges in the command line pre-built pipelines, workbenches such as Galaxy and Taverna provide the users to design step by step analyses just by a simple drag and drop functions available in these graphical interfaces. Galaxy and Taverna are two most popular bioinformatics server workbenches. Galaxy is a web-based interface, whereas Taverna is a stand-alone client which provide access to all the tools available freely on internet. Both can be installed locally and have the feature of sharing the workflow. (Leipzig J., 2017) A new tool can be added to the Galaxy analysis environment by writing a configuration file which has all the specifications as to how to run the tool, input/output parameters etc. This makes the ideal for the users who are not very much familiar to the programming languages. In terms of reproducibility the Galaxy seems promising. (Goecks J. et. al, 2010)

Taverna workbench is free and can be downloaded. It is compatible with windows, mac and Linux OS. Through Taverna access is available to several different tools and resource that are freely available. The workflows are reusable, reproducible and can be shared with other users. Another mode of executing the workflows from Taverna is by the Taverna Serve. Lastly, Taverna can also be executed by the Taverna Lite installation, which allows the users to run the workflows through the web and also upload new workflows. The Taverna bioinformatics user community include transcriptomic, proteomic and metabolomics analysis. (Wolstencroft K. *et al.*, 2013)

Chapter 3



Docker4seq

Chapter 3

Docker4seq

3.1 Introduction

Reproducible Bioinformatics Project (RBP) is a concept based on three modules: docker4seq R package, dockers image, and 4SeqGUI. (Figure 3.1) As discussed in the above chapters the aim of RBP is to develop easy to use bioinformatics workflows fulfilling the ten rules by Sandve G. K. *et al.* 2013.

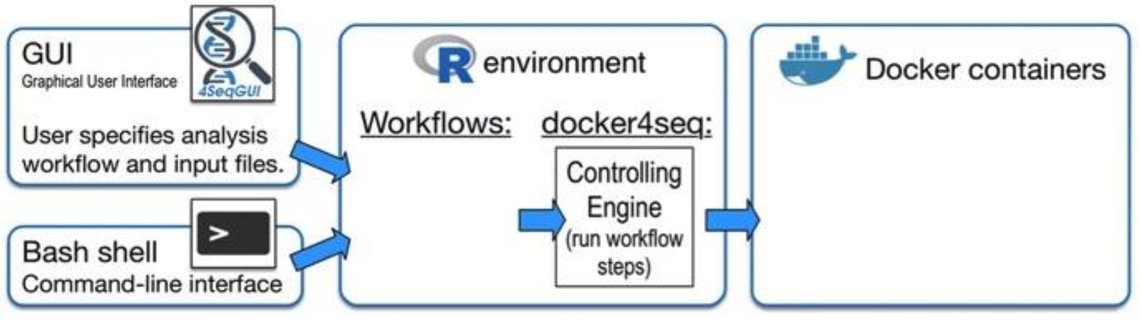


Figure 3.1: The Docker4seq package acts as an interface between users and docker containers.

R packages are a collection of R functions, complied code and sample data. They are stored under a directory called "library" in the R environment. By default, R installs a set of packages during installation. Docker4seq is a R package contains all the R functions which are required to handle all the steps of RNA-seq. miRNA-seq and ChIP-seq data analysis. The R package generate the commands required to execute docker containers capable of fulfilling tasks (e.g. short reads mapping, differential expression analysis, etc.). This approach provides multiple advantages such as following:

- The user has no need to install all the required software on its local server as they are all embedded into various containers
- Each container produces the results which are part of a specific pipelines
- By sharing the docker images used for the analysis, reproducible results are therefore guaranteed

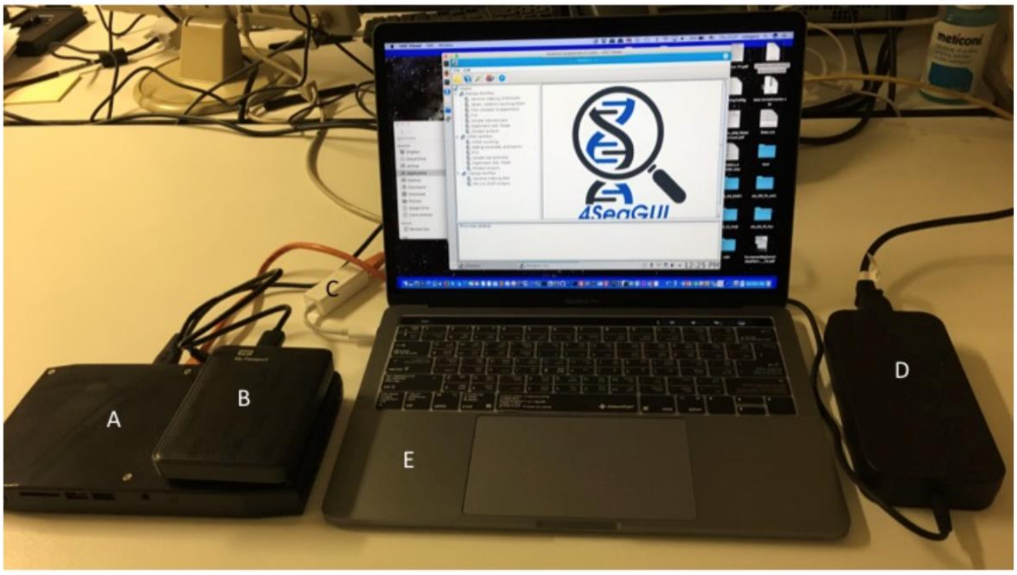


Figure 3.2: A – SeqBox; B – external hard disk; C – ethernet cable; D – SeqBox power supply; E – PC/MAC

An example of implementation of docker4seq is the SeqBox (Beccuti *et al.* 2018), an integrated hardware/software solution to facilitate data analysis to life scientists. Specifically the R engine, docker4seq, can be controlled by a graphical interface, 4SeqGUI, and the software is implemented in an Intel mini-computer equipped with 32GB GB RAM and 500GB Internal SSD. Docker4seq was built to provide a general schema and a software infrastructure to distribute robust and reproducible workflows. It provides the users the capability to repeat consistently any analysis independently by the UNIX-like architecture in use. The package can be downloaded and installed in R with the following commands:

```
library(devtools)
install_github("kendomaniac/docker4seq", ref="master")
library(docker4seq)
downloadContainers(group="docker")
```

Docker4seq is one of the packages being part of the Reproducible Bioinformatics Project (RBP), which is an open community for the development of reproducible workflows in bioinformatics. Developers can build their own packages and/or workflows in an R environment, which is the most widely-used programming language by scientists with different levels of scripting knowledge. The skeleton.R function embeds the runDocker function, which is the core of the construction of the string of text that will control a docker image. Actually, the runDocker function supports docker daemon, but we are extending it to support also singularity daemon (https://sylabs.io/docs/), which provides advantages such as security with respect to docker, since it allows to run docker containers not holding root privileges.

The skeleton.R function controls an ubuntu image, which is the prototype image to create a functional container in the docker image repository docker.io/repbioinfo. However, developers can make use of any linux image to build their own docker image, as there is no specific software requirement for the docker image present in the RBP. Any new workflow being part of RBP is required to fulfil at least the first 6 rules defined by Sandve G. K. *et al.*, 2013. Any new RBP module must be provided with an explanatory document in the form of vignette pages, i.e. an online html document, and a test data set.

3.1.1 Sequencing data types

Workflows of following of sequencing datatypes are implemented in docker4seq:

a) mRNAseq

RNA-seq, also rightly called as whole-transcriptome shotgun sequencing, it is used for high-throughput sequencing technologies to characterize the RNA content and composition in a given sample. The transcripts sequence information as full transcript cannot be retrieved, but it can be obtained in the form of short reads of up hundred base pairs. The starting point for biological inference, is obtaining by the read counts that fall onto a given transcript which finally provides a digital measurement of transcript abundance.

b) miRNAseq

The micro RNAs are a class on non-coding RNAs which have an important role to play in regulating the gene expression. The process of DNA transcribing into primary miRNAs and then further processed into precursor miRNAs and finally into mature miRNAs is well known (ref). Most of the times the miRNAs interact with 3’ untranslated regions (3’UTRs) of target mRNA to promote mRNA to degrade and translate repression (O'Brien J. *et al.*, 2018).

c) ChIPseq

The goal of ChIP-seq analysis is to determine how transcription factor and some other chromatin-associated protein influence phenotype-affecting mechanism. In short it provides insight into protein-DNA interactions for regulating gene expression, which is very important in understanding many biological process and disease states. (Pepke S. *et al.*, 2009)

3.1.2 Tools for the analysis of sequencing data

The analysis for the above data types is made using the software described below.

a) Data pre-processing

i. Demultiplexing: from bcl to fastq

The input used by downstream sequencing analysis applications are FASTQ files. The Illumina sequencing instruments generate per-cycle BCL basecall files as primary sequencing output. The bcl2fastq software combines these per-cycle BCL files from single run following to which it translates them into FASTQ files. bcl2fastq also separates multiplexed samples, this process of separating the sample is called demultiplexing. (bcl2fastq Conversion-User Guide, 2013) bcl2fastq requires that the information about the sequenced samples are provided by a comma separated file (SampleSheet.csv – Table 3.1)

Table 3.1: Sample sheet column header and description

Column	Description
FCID	Flow Cell ID
Lane	Positive integer, indicating the lane number (1-8)
SampleID	ID of the sample
SampleRef	The name of the reference
Index	Index sequence(s)
Description	Description of the sample
Control	Y indicates this lane is a control lane, N means sample
Recipe	Recipe used during sequencing
Operator	Name or ID of the operator
SampleProject	The project the sample belongs to

Every project in the sample sheet is linked to a corresponding project directory. Each sample belonging to that project is linked to a corresponding sample directory, within project directory. After their generation, FASTQ files are generated in the project and sample directories specified in the sample sheet.

The generation of FASTQ files, based on the index information, and production of sequencing run statistics is made by bcl2fastq and this project goes under the name of demultiplexing. Demultiplexing works in the following way:

1. Gets the raw index for each index read from .bcl files.
2. Identifies the appropriate directory for the index based on the sample sheet.
3. For each read:
   a) Write the index sequence into the index field.
   b) Append the read to the appropriate new FASTQ file in output directory.

ii. Technical quality control of fastq: FastQC & MultiQC

During sequencing, the nucleotide bases in a DNA or RNA sample (library) are determined by the sequencer. For each fragment in the library, a short sequence is generated, also called a read, which is simply the succession of nucleotides associated to the quality of the base calling. Modern sequencing technologies can generate a massive number of reads from a single experiment. However, no sequencing technology is perfectly efficient, and each instrument will generate different types and amounts of problems/errors, e.g. such as incorrect nucleotides being called.

Therefore, it is necessary to understand and identify errors that may affect downstream analysis. Thus, sequence quality control is therefore an essential first step in your analysis.

FastQC is a software used for quality check for high throughput sequencing data, developed by Simon Andrews at the Babraham Institute in Cambridge (https://www.bioinformatics.babraham.ac.uk/projects/fastqc/). It is an open-source tool that allows quality control of raw sequence data. It also allows a modular set of analyses, which can be uses to give a quick impression of whether data have any problem before doing any further analysis. FastQC looks at quality collectively across all reads within a sample, rather than looking at quality scores for each individual read (Wingett S. W. *et al.*, 2018).

MultiQC is a tool which provides visualization of the results generated by FastQC, trimming tools and mapping tools across all samples in a sequencing run or in an experiment. It is command line tool that you point at a directory containing output from your analysis. It runs through all of the files and finds output of most used tools for data processing in short reads sequencing, e.g. FastQC, SKEWER, STAR, etc. The analysis results are organized in a single HTML report summarising the QC for all samples in that specific experiment. This makes easy to spot any outliers and check trends. (Ewels P. *et al.*, 2016) In MultiQC12 programs output are currently supported (see below). Output from any of these modules is combined into the final report (Figure 3.3)

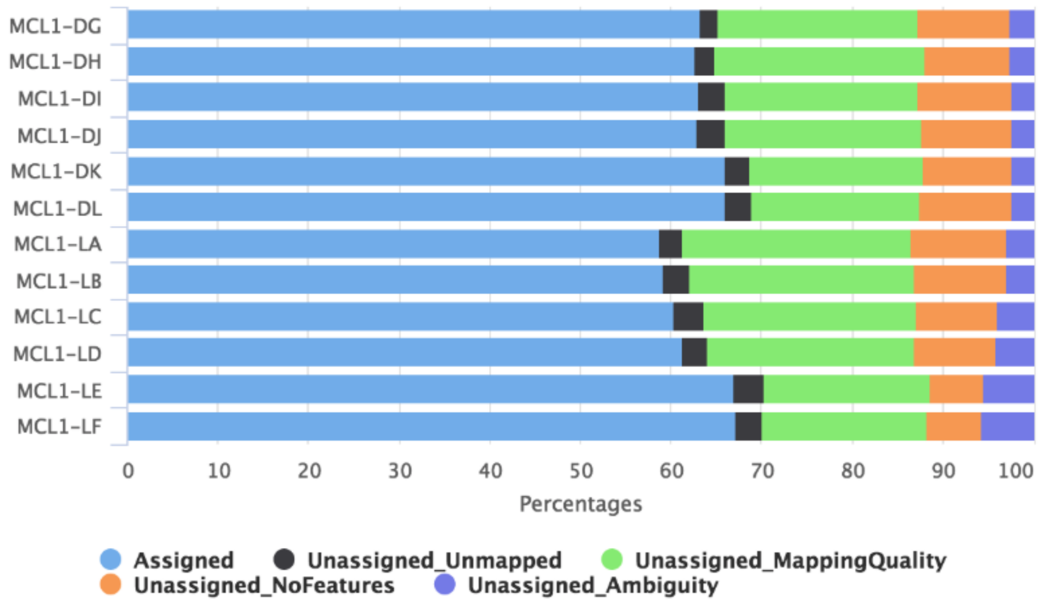


Figure 3.3: Example of stack chart for feature counts in multiQC report

iii. Adapter trimming

1) SKEWER

When the reads are longer than the target DNA/RNA fragments, adapter trimming becomes a prerequisite step before mapping with tools which are unable to perform soft-clipping, i.e. not considering in alignment 3' end region of a reads that do not map the targer sequence. Trimming is mandatory in docker4seq miRNA workflow because mapping is done using BWA, allowing only one mismatch between the read and the miRNA precursor sequence.

SKEWER is a trimming program designed to process Illumina sequences and it is based on a dynamic programming algorithm, specifically dedicated to the task of adapter trimming (Jiang H. *et al.*, 2014). Skewer is time efficient and has the best capabilities of achieving unmatched accuracies for adapter trimming. SKEWER algorithm searches for adapter sequence pattern in an exhaustive and efficient manner. It is designed in such a way that it can be used in both SE and PE sequencing data. Jiang H. *et al.*, 2014 have provided the results for Skewer tool and was shown to achieve accuracies that were not matched by other similar tools that are currently available. Importantly, Skewer is optimized for most applications. (Figure 3.4)

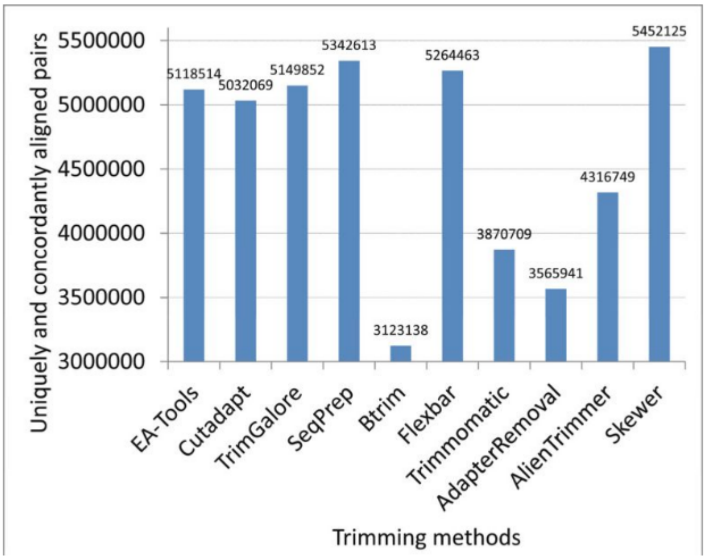


Figure 3.4: Bar chart showing the performance of different trimming tools

2) Cutadapt

Trimming is a mandatory procedure in miRNAseq and Cutadapt was described specifically as trimming tool for miRNA sequencing. Cutadapt works in error-tolerant way as it can identify, modify and filter reads in various ways (Figure 3.5). It works step wise identifying and removing adapter sequences, primers, poly-A tails. It can either trim or discard reads in which an adapter occurs and can also discard reads that are below a specified length after trimming. (Martin M., 2010)

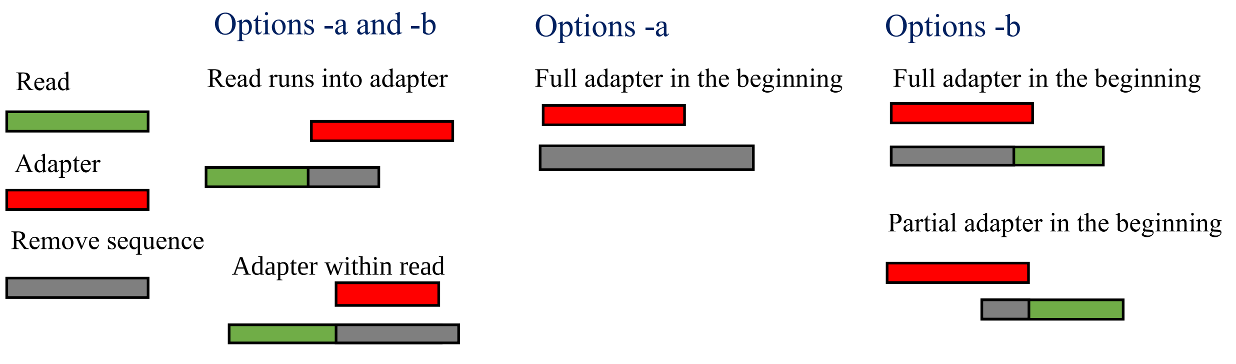


Figure 3.5: Schematic representation of Cutadapt algorithm: possible configuration between the read and an adapter sequence. The two -a and -b are used as adapater trimming options.

b) Mapping tools

i. Genome mapping (alignment-based mapping)

Reads that are obtained from the sequencer are ideally mapped to either genome or transcriptome. An important part of mapping is the percentage of mapped reads, which indicates the overall sequence accuracy and the presence of contamination. It is expected that the reads of a regular RNA-seq maps closed 90% onto its reference genome. On the other hand, when the reads are mapped against the transcriptome the percentage of mapped reads is expected to be slightly low, as the reads coming from the unannotated region might also be lost (Conesa A. *et al.*, 2016).

1) STAR

The alignment of the reads to the reference genome is done using STAR (Spliced Transcripts Alignment to a Reference, Dobin A. *et al.*, 2013), to determine where on the reference genome the reads are originated. Many challenges of RNA-seq data mapping is handled by STAR aligner using a strategy to account for spliced alignments.

For every read that STAR aligns, STAR will search for the longest sequence that exactly matches one or more locations on the reference genome. These longest matching sequences are called the Maximal Mappable Prefixes (MMPs): The different parts of the read that are mapped separately are called ‘seeds’. So, the first MMP that is mapped to the genome is called seed1. STAR will then search again for only the unmapped portion of the read to find the next longest sequence that exactly matches the reference genome, or the next MMP, which will be seed2. (Dobin A. *et al.*, 2013)

This sequential searching of only the unmapped portions of reads underlies the efficiency of the STAR algorithm. STAR uses an uncompressed suffix array (SA) to efficiently search for the MMPs, this allows for quick searching against even the largest reference genomes. Other slower aligners use algorithms that often search for the entire read sequence before splitting reads and performing iterative rounds of mapping. If STAR does not find an exact matching sequence for each part of the read due to mismatches or indels, the previous MMPs will be extended. (Dobin A. *et al.*, 2013)

If extension does not give a good alignment, then the poor quality or adapter sequence (or other contaminating sequence) will be soft clipped. The separate seeds are then stitched together to create a complete read by first clustering the seeds together based on proximity to a set of ‘anchor’ seeds, or seeds that are not multi-mapping. Then the seeds are stitched together based on the best alignment for the read (scoring based on mismatches, indels, gaps, etc. (Dobin A. *et al.*, 2013)

2) Burrows-Wheeler Aligner (BWA)

To align low divergent sequences to a reference genome or not spliced transcriptome BWA is used. There are three algorithms in the BWA package such as BWA-backtrack, BWA-SW and BWA-MEM. BWA-backtrack is used for aligning reads up to 100bp, while the other two are used for the longer sequences ranging from 70bp to 1Mbp, they also provide split alignment. BWA provides and output of the alignment in the new standard format called Sequence Alignment/Map (SAM). These SAM files can be further used for analysis using open source SAM tool packages. What is more, BWA provides support to the pair-end read mapping, by identifying the position of all the good hits, sorts them by the chromosomal co-ordinates finally scans through all the hits to pair the two partner ends. (Li H. and Durbin R., 2009)

3) SHRiMP

The identification of mature miRNAs can be done using as reference the hot genome or miRNA precursors. miRbase database consists on the predicted hairpin portion of the miRNA transcript along with the information of its location and mature miRNA sequence (termed as miR). The hairpin and mature miRNA sequence can be searched and browsed within the database. The sequences and annotation of the data are both present in the database to download. Since the miRbase is a huge repository for miRNA sequences, it is used for the mapping the unknown miRNAs to miRbase database. (Griffiths-Jones S. *et al.*, 2006) SHRiMP is based on q-gram filter approaches (Rasmussen R. K. *et al.*, 2006), spaced seeds (Califano A. and Rigoutsos I., 1993) and an optimized Smith-Waterman Algorithm (Rumble S. M. *et al.*, 2009) (Figure 3.6)

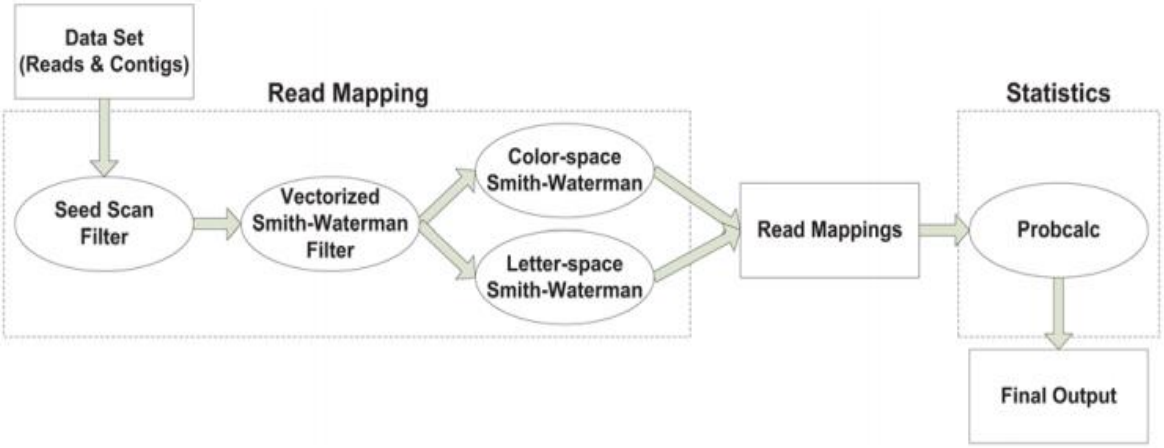


Figure 3.6: SHRiMP Data flow and processing

ii. Quasi-alignment (alignment-free mapping)

Quasi-alignment is an efficient mapping algorithm which maps the sequencing read to the transcriptome, by attempting to report potential loci of the origin of the sequencing read and merely does not provide base to base alignment of sequencing read to reference. Quasi mapping is considered to be faster in terms of speed as compared to the other exiting alignment tools (Srivastava A. *et al.*, 2016). Moreover, in case of the alignment to the transcript rather than the genome, there are a lot of repetitive mapping, as alternative splicing give rise to many identical alignments for reads, which means that the reads might align to different transcript and different position, this adds to a lot of redundant work of solving the identical alignment. In docker4seq, Salmon is implemented as quasi-alignment tool, to provide the possibility to perform mapping on computers with relatively low RAM, i.e. < 32 GB, which are instead mandatory for STAR mapping on large genomes as the human one.

SALMON

SALMON is a transcript quantification tool. To quantify the reads the tool requires a set of target transcripts, could be either from reference or de-novo assembly. The basic requirement for using SALMON is a FASTA file containing the reference transcripts and a set of FASTA or FASTQ files having reads. The quasi-mapping-based mode of Salmon runs in two phases; indexing and quantification. The indexing step is independent of the reads to be used for quantification, and only need to be run ones for a particular set of reference transcripts. The quantification step is specific to the set of RNA-seq reads (Patro R. *et al.*, 2017).

c) Read counting tool

RSEM

In transcript quantification from RNA-Seq data, the challenge is the handling of reads that can map to multiple genes or isoforms. In cases where the sequenced genomes are absent, issue is particularly important for quantification with de novo transcriptome.

RSEM is a software package, which is used for quantifying gene and isoform abundances from RNA-seq data for both single-end and paired-end data. RSEM outputs gene and isoform abundance estimates. RSEM does not require a reference genome for transcript expression estimation. In cases a de novo transcriptome assembler is combined with RSEM can provide accurate transcript quantification for species without sequenced genomes. Tests on real data sets, showed that RSEM has better or similar performance to other quantification techniques that heavily rely on a reference genome.

RSEM consists of two steps:

1. The first and most important step for RSEM to generate a set of reference transcript sequences and pre-processed it for later use by RSEM steps
2. Further the second step is an alignment to reference transcript is done to a set of RNA-seq reads, the resulting aligned reads are used to estimate abundances. The scripts rsem-prepare-reference and rsem-calculate-expression are the scripts which carries out these two steps. (Bo Li and Dewey C. N., 2011)

d) Gene annotation using ENSEMBL

Gene annotation is nothing but providing a link between HUGO symbols and reference data base ids. In docker4seq we have decided to use ENSEMBL database as source of annotation and genome assembly essentially because:

- ENSEMBL is gene centric. Thus there is no needs to query other database to associate transcrpts to genes as instead is required for Refseq and UCSC databases.
- Any release version of the ENSEMBL is stored in their ftp repository, which provide the basis for the reproducibility of the annotation. E.g. UCSC do not have stored version of the annotation and annotation GTF is created using a GUI and there is no way to be sure that a GTF created in an other moment will contain the same information.
- ENSEMBL has the most annotated non-coding genes from a wide variety of organisms.

e) Data reformatting

In docker4seq the output of each step of the analysis is compatible with the following one. STAR/RSEM and Salmon were configured to produce the same files in structure and content to be used by a specific R function (samples2exeriment), which aggregate the outputs of the above mapping tools to provide counts, TPM, FPKM tables for genes and transcripts to be used for differential exspression analysis.

f) Differential Expression analysis

Differentially expression analysis is a statistical analysis necessary to extrapolate, from a transcriptomic experiment, which are the genes/transcripts involved in a specific biological event. There are many different differential expression analysis tools such as DESeq, DESeq2, EdgR, Limma, NBPSeq etc which work specific to the data selected. In docker4seq we have implemented DESeq2 and edgeR. DESeq2 has a very high sensitivity and precision compared to edgeR and voom (ref), and it has features, which are not available in other tools. One of these features is, the Empirical Bayes shrinkage for FC (fold change), which shrinks log FC estimates toward zero. This feature reduces the noise due to low expressed genes, since shrinkage is stronger when the available information for a gene is low, which may be because the read counts are low. Another feature of DESeq2 is the identification of counts outliers, which could lead to false positives. Specifically, DESeq2 flags genes that record counts outliers, estimated with the standard outlier diagnostic Cook’s distance. (Soneson C. and Delorenzi M., 2013)

1) DEseq2

The DEseq2 an R package embedded and implemented to perform differential expression analysis. DEseq2 expects the data to be count matrix for mRNA and miRNA seq data. The count data are presented in the form of a table which reports for each sample, the number of sequence fragments that are assigned to each gene. DEseq2 R package allows methods to validate the differential expression by using the negative binomial generalized models. The estimates of dispersion and the log 2-fold changes include data driven prior distribution. The vignette pages are explained in details about the use of packages and demonstrate typical workflows. The vignette page on the Bioconductor incorporates the RNA-seq workflow which provides similar material of generating count matrices from FASTQ and following analysis including downstream analysis. (Love M. I. *et al.*, 2014).

A comparison study was conducted to test the performance of different differentially expression analysis tools such as ALDEx, DESeq, edgeR, and CuffDiff. Paper results suggest that DESeq is one of the most robust algorithms, hence preferred in many studies (Fernandes A. D. et. al, 2013)

2) Anova-like

ANOVA-like DE procedure is to identify the genes with larger difference between the conditions and within condition difference. It has the capabilities to show differential expression and magnitude of the comparative difference even in a very small sample size, and provides a multiple contrast of conditions. To offer docker4seq users a multi-group comparison, we have implemented the Anova-like statistics present in edgeR. (Fernandes A. D. et. al, 2013)

g) Peak calling

Peak calling is an important step in ChIP-seq analysis. It defines the protein-DNA binding region by identifying the region where most of the reads mapped to the genome. Peak caller tools use the following steps:

a. *Read shifting:* Most of the ChIP-seq data comes from single-end sequencing. The reads are aligned to the sense/antisense strand of the genome, they are shifted and combined for obtaining a ratio close to 1, which is used to identify protein binding regions.

b. *Background estimation:* Control ChIPs are also processed in the same way to allow either a genomic background to be determined or for regions enriched through the ChIP process with no antibody specificity to be identified (IgG controls).

c. *Peak identification:* peaks are identified by the reads aligned to the particular region above the threshold or it is highly enriched in the aligned region compared to the background signal.

d. *Significance analysis:* The peak significance is determined by the statistical method and takes into consideration the p value or FDR; for statistical analysis the true peak list is necessary for comparison. Few tools take into consideration the height of peaks and/or enrichment with background to rank peaks, however this method does not provide p values.

e. *Artefact removal:* Before generating the final peak list, the peaks in which contains either single read or few reads are discarded as this can be due to the PCR amplification artefacts. Also, the peaks which are significantly difference between the number of reads on each strand are also discarded.

MACS

There are many peak calling algorithms. MACS is one of the most popular algorithms and widely used by many researchers to detect peaks, therefore MACS was implemented in the docker4seq package. In a study by Elizabeth G, et.al, (2010), eleven different peak calling algorithms were tested on three different datasets and their sensitivity, accuracy and usability were evaluated. MACs was identified as one of the most robust algorithms in peak detection across the three dataset considered in the study (Figure 3.8).

h) Data visualization

PCA

PCA is widely-used for data visualization of miRNA and mRNA data analysis. The principal component analysis (PCA) is a dimensionality reduction method based on variance of the data. This method converts a set of observations of correlated variables into a set of values of linearly uncorrelated variables called principal components. By using few components from the data, each sample can be represented by relatively few numbers instead of using the values for thousands of variables. Finally, the data can be visualized and similarities and differences between samples can be easily plotted with PCA.

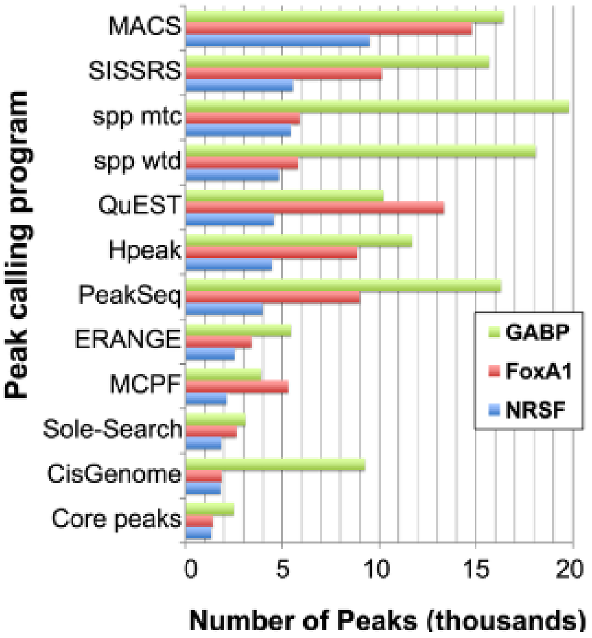


Figure 3.8: Peaks identification **via** different algorithms. GABP, FoxA1 and NRSF are three different xdatasets. (Elizabeth G. Wilbanks, 2010)

3.2 Docker4Seq workflow strategy

There are numerous tools and packages available for the analysis of different types of sequencing data, but these are often individually applied and there is no current platform implementing all of them under one roof. Therefore, I have developed a user-friendly workflow that can make use of different bioinformatics tools to obtain numerous results on the same data sets. All workflows embedded in docker4seq are characterized by providing, for each module, a output file with is compatible for the next step of the analysis. This is particularly important to avoid manual manipulation of the data (Sandve G. K. *et al.*, 2013).

The strategy implemented is based on an initial assessment of the data quality, the removal of any adapter sequences and alignment to the corresponding genome/transcriptome. These steps can be applied mRNA, miRNA and ChIP-seq, while subsequent analysis steps have been individually tailored for each of the three workflows. These three data types were chosen for both the development and validation of the workflows, because they are the most widely-used sequencing data types and the designed tool would be useful for a larger number of users, including biologists with no prior programming skills.

The analysis of all three workflows starts with the demultiplexing of bcl input files and their conversion to fastq files using bcl2fastq, which is an optional step, if the inputs are already in fastq format. The quality of fastq files is then checked using FastQC and trimmed using either SKEWER and CutAdapt, depending on the data type. The trimmed files are then fed to the alignment tools, which can again vary depending on the workflow.

1. Analysis of mRNAseq

Based on the choice of the user, the mRNA-seq workflow implementation in the docker4seq package provides two options:

1) a genome based alignment approach using STAR (Dobin A. *et al.*, 2013) and RSEM (Zhang C. *et al.*, 2017)
2) a transcriptome-based approach based on the quasi-alignment tool Salmon.

The reason of the implementation of two different mapping tools is to provide higher flexibility to the final user in terms of specificity and different hardware requirements. STAR, although being memory intensive, it is shown to have high accuracy and outperforms other available aligners by more than a factor of 50 in mapping speed, hence I have implemented

STAR as an aligner in docker4seq package. The algorithm achieves this highly efficient mapping by performing a two-step process such as a) Seed searching and b) Clustering, stitching, and scoring.

SALMON was embedded in the docker4seq for presenting the following advantages:

- Fast & Lightweight – can quantify 20 million reads in under eight minutes on a desktop computer
- Support for strand-specific libraries
- Accepts BAM and FASTQ files as input
- Lower RAM requirements with respect to STAR/RSEM quantification approach

Zhang and co-workers performed a comparative analysis between alignment dependent tools such Salmon_aln, eXpress, RSEM, TIGAR2 and quasi-alignment methods such Salmon, Kallisto, Sailfish was performed (BMC Genomics 2017, 18,583). STAR was used as mapping tool for alignment-dependent methods. For the isoform quantification, the authors indicated there was a strong co-relation among the quantification results obtained from RSEM, Salmon, Salmon_aln, Kallisto and Sailfish with R2 > 0.89, indicating that impact of mappers on isoform quantification is small. For the gene-level quantification, the difference between quasi-alignment methods and alignment-dependent tools is not as significant as for transcript level analysis (Teng M. *et al.*, 2016). However, quasi-mapping methods require lower memory, being a good alternative to genome mapping tools, and thus we also implemented SALMON in docker4seq.

The bam outputs aligned with STAR are subsequently parsed to RSEM for generating a count table and the results are sample specific. The same output is also generated by Salmon, to keep output downstream compatibility. RSEM or Salmon count tables are then assembled in an experiment table, which also include covariates and batch information, whenever is needed. Experiment tables are provided as counts for differential expression and as TPM/FPKM format for data visualization, e.g. PCA or heatmaps. As indicated above counts experiment table is then used for differential expression analysis, which can be performed through DESeq2 and Anova-like methods, chosen for their increased sensitivity for different RNA data sets. Outputs of differentially expressed tools have again the same format and it can be used by a filtering function to generate set of counts/TPM/FPKM tables filtered for the set of the differentially expressed genes to be used for the heatmap module implemented in docker4seq or to be used with other external clustering tools.

2. miRNAseq

Cordero F. *et al.* 2012 had designed, optimized and validated a miRNA workflow, which I have used as reference for building a new, independent miRNA workflow in the Reproducible Bioinformatics Project. As mentioned above, miRNA data sets were analysed in a similar manner to mRNA datasets, starting from bcl inputs conversion to fastq and quality check via FastQC. However, the adapter sequences were removed using CutAdapt, which was proven to be more efficient when reads are longer than the molecule sequenced, such as in microRNA data. Subsequently, the filtered fastq files were mapped using Shrimp, instead of STAR, because Shrimp takes into consideration smaller read lengths and uses local alignment method based on Smith-Waterman Algorithm, which increases precision.

Since same mature miRNA can be generated by multiple miRNA genes dispersed over the genome, some of the reads could be lost as multiple mapping, when using the genome as reference for mature miRNA (□20 nts) search. Furthermore, mapping approximately 20 nts over 3 billion bases of high eukaryotes genomes, could result in fake mapping, because of the possibility to detect similarities with potential miRNA targets in the 3’ end of genes. For this reason, miRNA precursors from miRbase database were mapped using Shrimp.

The quantification of the miRNA expression was then performed using two methods - annotation based and the position based. In the annotation-based method, reads mapping on precursors weree saved in a genomicRanges object, which was compared to the miRbase annotated location of mature miRNA in its precursors. mirBase mature location in precursor was also integrated with positional information, when the localization of the mature miRNA in the 3’ end (-3P) or in the 5’ end (-5P) of the precursor was not indicated in miRbase.

Finally, a count matrix containing the merged results of the two methods of miRNA quantification was generated. The annotated positions of the mature miRNAs were indicated in the final table with the postfix “:Novel” instead of “:mirBase”.

Following mapping and quantification, mirnaCovar() was used to add the covariates and batch information for the miRNAseq raw counts. The resulting experimental table was then used as input for differential expression analysis, which could be performed using DESeq2 or Anova-like methods, as previously described for mRNA data sets.

Lastly, further analysis such as evaluation of the experiment power and sample size, as well as visualization using PCA are available both for mRNA and miRNA workflows.

3. ChiPseq

Pre-processing of CHiPseq data sets was done similarly to that for mRNAs, such as demultiplexing, quality check via FastQC and adapter trimming using SKEWER. Subsequently, the filtered reads were aligned using BWA-MEM, known for its high-quality performance and accuracy fit for CHiPseq data sets. MACS was then used for peak calling, as it is considered the most robust and reliable algorithm. The final peak list obtained was annotated through a custom script using genomic Ranges and refGenome R package to allocate peaks with respect to the nearest genes.

The structure of the ChIP-seq workflow was based on the previous work of Dr. Matteo Carrara, who embedded it in basespace. The ChIP-seq workflow was however, implemented in the docker4seq infrastructure by myself.

3.3 Results

As mentioned above, the RBP described within this project is unique for the implementation of all three components – Docker4Seq, Docker containers and 4SeqGUI. It integrates command-line (Bash) and R programming languages with a **user-friendly** GUI interface, making this tool/workflow accessible for both experienced bioinformaticians, as well as biologists with little-to-no prior skills in computational analysis. In addition to this, it confers **flexibility** for optimising and further development of individual tasks and/or parameters for specific containers. By applying the strategy discussed above for the three types of sequencing data sets, I have developed a workflow/pipeline consisting of consecutive functions described in details, including their corresponding commands, parameters and options that the user can follow step-by-step and implement according to their proposed analyses. (Figure 3.9)

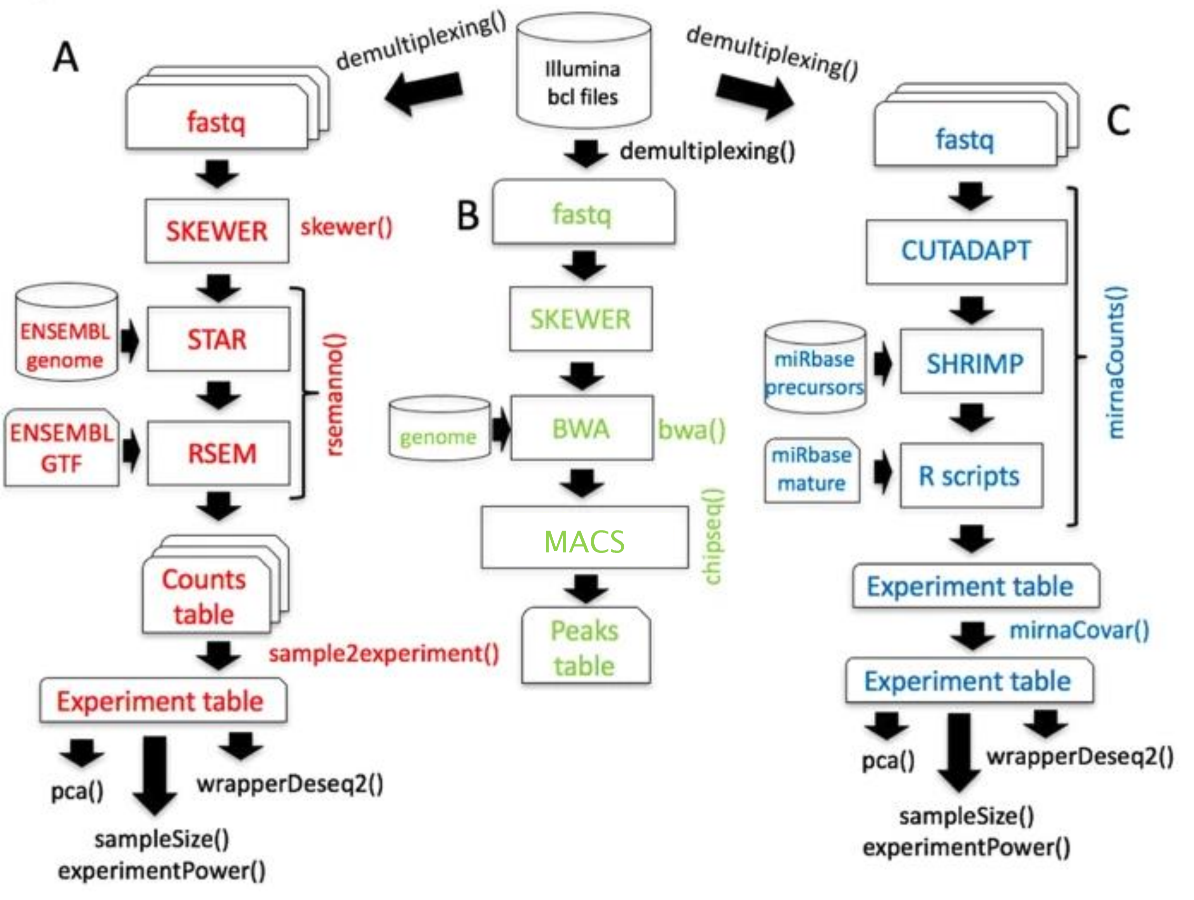


Figure 3.9: Flow chart of different workflows implemented in Docker4seq: mRNA (A), ChIP (B) and miRNA (C)

The Docker4seq has been organized in two branches one is the stable and the other one is the development branch. The transition from development to stable is when a module made of R function and docker container satisfies the 10 rules proposed by the Sandve G. K., *et al.*, 2013. We have used Git and GitHub as it useful when developing a package. An R user can make an easy installation using the same package by using following two lines of code.

```
install.packages("devtools")
devtools::install_github("username/packagename")
```

The nomenclature of docker images are labelled with an extension YYYY.NN, wherein YYYY stands for the year of insertion in the stable version, while NN is a progressive number. There can be a change in the YYYY only when the there is an update done in the programmes, implemented in the docker image, as any such updates will affect the reproducibility of the workflow in RBP. The previous versions will remain available in the repository. As mentioned earlier the NN signifies the change in the docker image, which will not affect the reproducibility in those workflows in RBP.

All the R functions necessary in handling all the steps of RNA-seq, ChIP-seq and miRNA-seq workflows are embedded in the stable branch of docker4seq R package. At the present time all functions requiring calculations are embedded in the following docker images:

docker.io/repbioinfo/demultiplexing.2017.01 used by demultiplexing

docker.io/repbioinfo/annotate.2017.01 used by rnaseqCounts, rsemanno

docker.io/repbioinfo/bwa.2017.01 used by bwaIndexUcsc, bwa, wrapperPdx

docker.io/repbioinfo/chipseq.2017.01 used by chipseqCounts, chipseq

docker.io/repbioinfo/r332.2017.01 used by experimentPower, sampleSize, wrapperDeseq2

docker.io/repbioinfo/mirnaseq.2017.01 used by mirnaCounts

docker.io/repbioinfo/rsemstar.2017.01used by rnaseqCounts, rsemstarIndex, rsemstarUscsIndex

docker.io/repbioinfo/skewer.2017.01 used by skewer, rnaseqCounts, wrapperPdx

docker.io/repbioinfo/xenome.2017.01 used by xenome, xenomeIndex, wrapperPdx

Any analysis performed with docker4seq generate a file called containers.txt, which docker images are available in the local release of docker4seq. In case, user would like to download a set of docker images different from those provided as part of the package, then these images must be specified in a file with the format docker.repository/user/docker.name, which has to be passed to downloadContainers function:

```
downloadContainers(group="docker", containers.file="my_containers.txt")
An example of the my_containers.txt file content is:
docker.io/repbioinfo/bwa.2017.01
docker.io/repbioinfo/chipseq.2017.01
docker.io/repbioinfo/r340.2017.01
```

R skeleton for developing workflows in RBP

- The first step in the skeleton function is storing the working folder and grabbing the process time for subsequent performance evaluation.
- Testing if docker demon is running.
- Setting the working directory to the data folder.
- Checking if the scratch folder exists and then creating a temporary folder
- Executing the docker command:

```
skeleton(group="docker", scratch.folder, data.folder)
```

The **skeleton.sh** scripts in docker.io/repbioinfo/Ubuntu for eg: write the hello world in the **helloworld.txt** and moves **helloworld.txt** to the scratch folder together with the **run.info** file, which is used to store information about the run, and the **out.info**, used to tell to the R script when the doker job is finished. The **skeleton.sh** scripts is a prototype for the handling of docker application(s).

The **resultRun** is used to check when the docker job is finished. The results are copied from the scratch to the data folder, followed to which the computing time is estimated and saved in the run.info file. The log of the docker job (describing the events observed during the execution of the container) is saved with a name made of the first 12 letters of the docker job ID. After saving the docker instance log, temporary directories and files are deleted.

More details about the R skeleton and a step-by-step tutorial can be found at: https://kendomaniac.github.io/docker4seq/articles/skeleton.html

Workflow for data pre-processing

Demultiplexing, quality check and adapter trimming is performed for all sequencing data types such as mRNA, miRNA and CHiPseq, but are not implemented in 4SeqGUI, because these steps are usually performed by a core lab. Thus only a limited group of users require the use of this function. The mRNAseq workflow, that can be handled using 4SeqGUI graphical interface (Linux/MAC), starts from the availability of fastq files. Therefore, they can be executed from the command line, as follows:

1. **Demultiplexing**

```
demultiplexing(group="docker", data.folder, scratch.folder, threads=24)
```

2. **Quality check using FastQC**

```
fastqc(group = c("sudo", "docker"), data.folder)
```

User has to create the **fastq.folder**, where the fastq.gz files for all miRNAs under analysis are located. The **scratch.folder** is the location where temporary data are created. The results will be then saved in the **fastq.folder**.

3. **Adapter trimming**

- using either SKEWER

```
skewer(group = c("sudo", "docker"), fastq.folder = getwd(),
        scratch.folder = "/data/scratch", adapter5, adapter3,
        seq.type = c("se", "pe"), threads = 1, min.length = 18)
```

User needs to provide the adapter sequence of the sequencing adapters, adapter5 and adapter3 parameters. The min.length refers to the minimal length that reads should have after adapters trimming. Since today the average read length for a RNAseq experiment is 50 or 75 nts then it would be better to bring to 40 nts the min.length parameter to increase the precision in assigning the correct position on the genome.

- or CutAdapt

```
cutadapt(group = c("sudo", "docker"), scratch.folder, data.folder, adapter.type
= c("ILLUMINA", "NEB"), threads = 1)
```

More details and step-by-step guidelines can be found at:
https://kendomaniac.github.io/docker4seq/articles/docker4seq.html#rnaseq-workflow-howto

1. Workflow for mRNAseq analysis

The main tasks used for mRNA analysis can be setup using 4SeqGUI or command-line:

1.1 Creation of genome index for STAR (hg38 example below)

```
rsemstarIndex(group="docker", genome.folder="/data/scratch/hg38star",
ensembl.urlgenome="ftp://ftp.ensembl.org/pub/release-87/fasta/homo_sapiens/dna/
Homo_sapiens.GRCh38.dna.toplevel.fa.gz",
ensembl.urlgtf="ftp://ftp.ensembl.org/pub/release-87/gtf/homo_sapiens/Homo_sapi
ens.GRCh38.87.gtf.gz")
```

User has to provide the URL (ensembl.urlgenome) for the file XXXXX_dna.toplevel.fa.gz related to the organism of interest, the URL (ensembl.urlgtf) for the annotation GTF XXX.gtf.gz and the path to the folder where the index will be generated (genome.folder).

1.2 Mapping of reads to a reference genome using STAR (hg38 example below)

```
rsemstar(group=c("sudo","docker"),fastq.folder=getwd(), scratch.folder="/data/s
cratch", genome.folder, seq.type=c("se","pe"), strandness=c("none","forward","r
everse"), threads=1, save.bam = TRUE){--outSAMattributes Standard
```

1.3 Count generation using RSEM

```
#test example
system("wget http://130.192.119.59/public/test.mrnaCounts.zip")
unzip("test.mrnaCounts.zip")
setwd("./test.mrnaCounts")
library(docker4seq)
rnaseqCounts(group="docker",fastq.folder=getwd(), scratch.folder=getwd(),
adapter5="AGATCGGAAGAGCACACGTCTGAACTCCAGTCA",
adapter3="AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT",
seq.type="se", threads=8,  min.length=40,
genome.folder="/data/scratch/mm10star", strandness="none", save.bam=FALSE,
org="mm10", annotation.type="gtfENSEMBL")
```

1.4 Reference-free alignment and quantification of gene and transcripts using Salmon

```
#running salmonIndex human
salmonIndex(group="docker", index.folder=getwd(),
ensembl.urltranscriptome="ftp://ftp.ensembl.org/pub/release-90/fasta/homo_sapie
ns/cdna/Homo_sapiens.GRCh38.cdna.all.fa.gz", ensembl.urlgtf=ftp://ftp.ensembl.o
rg/pub/release-90/gtf/homo_sapiens/Homo_sapiens.GRCh38.90.gtf.gz, k=31)

#running salmonCounts
wrapperSalmon(group="docker", scratch.folder="/data/scratch/",
 fastq.folder=getwd(), index.folder="/data/genome/salmonhg38/",
 threads=8, seq.type="pe", adapter5="AGATCGGAAGAGCACACGTCTGAACTCCAGTCA",
 adapter3="AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT", min.length=40,strandness="none")
```

1.5 Generation of count matrix

RSEM and Salmon quantification generate counts, TPM and FPKM values for individual samples. However, I developed a sample2experiment function, which builds a matrix of counts and adds batch information for all samples that can be used in downstream analysis.

```
#test example
system("wget http://130.192.119.59/public/test.samples2experiment.zip")
unzip("test.samples2experiment.zip")
setwd("test.samples2experiment")
library(docker4seq)
sample2experiment(sample.folders=c("./e1g","./e2g","./e3g",
"./p1g", "./p2g", "./p3g"),
covariates=c("Cov.1","Cov.1","Cov.1","Cov.2","Cov.2","Cov.2"),
bio.type="protein_coding", output.prefix=".")
```

1.6 Evaluating sample size and experiment power

```
#test example
system("wget 130.192.119.59/public/test.analysis.zip")
unzip("test.analysis.zip")
setwd("test.analysis")
library(docker4seq)
sampleSize(group="docker", filename="_counts.txt", power=0.80, FDR=0.1, genes4d
ispersion=200, log2fold.change=1)
```

The requested parameters are the path to the counts experiment table generated by **samples2experiment** function. The param **power** indicates the expected fraction of differentially expressed gene, e.g 0.80. **FDR** and **log2fold.change** are the two thresholds used to define the set of differentially expressed genes of interest. The output file is **sample_size_evaluation.txt** is saved in the R working folder.

```
#test example
system("wget 130.192.119.59/public/test.analysis.zip")
unzip("test.analysis.zip")
setwd("test.analysis")
library(docker4seq)
experimentPower(group="docker", filename="_counts.txt", replicatesXgroup=7, FDR
=0.1, genes4dispersion=200, log2fold.change=1)
```

The requested parameters are the path to the counts experiment table generated by **samples2experiment** function. The param **replicatesXgroup** indicates the number of sample associated with each of the two covariates. **FDR** and **log2fold.change** are the two thresholds used to define the set of differentially expressed genes of interest. **genes4dispersion** indicates the number of genes used in the estimation of read counts and dispersion distribution. The output file is **power_estimation.txt**.

1.7 Differential expression analysis using DESeq2

```
#test example
system("wget 130.192.119.59/public/test.analysis.zip")
unzip("test.analysis.zip")
setwd("test.analysis")
library(docker4seq)
wrapperDeseq2(output.folder=getwd(), group="docker", experiment.table="_counts.
txt", log2fc=1, fdr=0.1,  ref.covar="Cov.1", type="gene", batch=FALSE)
```

1.8 Data visualization using PCA

```
#test example
system("wget 130.192.119.59/public/test.analysis.zip")
unzip("test.analysis.zip")
setwd("test.analysis")
library(docker4seq)
pca(experiment.table="_log2FPKM.txt", type="FPKM", legend.position="topleft", c
ovariatesInNames=FALSE, principal.components=c(1,2), pdf = TRUE, output.folder=
getwd())
```

More details and step-by-step guidelines can be found at:

https://kendomaniac.github.io/docker4seq/articles/docker4seq.html#rnaseq-workflow-howto

2. Workflow for miRNAseq analysis

In brief, fastq files are trimmed using CutAdapt and the trimmed reads are mapped on miRNA precursors, i.e. harpin.fa file, from miRBase using SHRIMP. Using the location of the mature miRNAs in the precursor, countOverlaps function, from the Bioconductor package GenomicRanges is used to quantify the reads mapping on mature miRNAs. All the functions for miRNA analysis can be performed via the 4SeqGUI interface.

User has to provide also the identifier of the miRBase organism, e.g. **hsa** for Homo sapiens, **mmu** for Mus musculus. If the **download.status** is set to FALSE, mirnaCounts uses miRBase release 21, if it is set to TRUE the lastest version of precursor and mature miRNAs will be downloaded from miRBase.

The main tasks implemented for miRNA data set analysis involve:

2.1 miRNA counting

```
#test example
system("wget 130.192.119.59/public/test.mirnaCounts.zip")
unzip("test.mirnaCounts.zip")
setwd("test.mirnaCounts")
library(docker4seq)
mirnaCounts(group="docker",fastq.folder=getwd(), scratch.folder="/data/scratch"
,mirbase.id="hsa",download.status=FALSE,adapter.type="NEB",trimmed.fastq=FALSE)
```

2.2 Addition of covariates and batch information

The function **mirnaCovar** is used to add to the header of all.counts.txt covariates and batches or covariates only. The output of **mirnaCovar**, i.e. w_covar_batch_all.counts.txt, is compliant with downstream analysis.

```
#test example
system("wget 130.192.119.59/public/test.mirna.analysis.zip")
unzip("test.mirna.analysis.zip")
setwd("test.mirna.analysis")
library(docker4seq)
mirnaCovar(experiment.folder=paste(getwd(), "all.counts.txt", sep="/"),
     covariates=c("Cov.1", "Cov.1", "Cov.1", "Cov.1", "Cov.1", "Cov.1",
                  "Cov.2", "Cov.2", "Cov.2", "Cov.2", "Cov.2", "Cov.2"),
     batches=c("bath.1", "bath.1", "bath.2", "bath.2", "batch.1", "batch.1",
               "batch.2", "batch.2","batch.1", "batch.1","bath.2", "bath.2"), o
utput.folder=getwd())
```

2.3 Differential expression analysis and data visualization were performed as before, for mRNA, described above.

More details and step-by-step guidelines can be found at: https://kendomaniac.github.io/docker4seq/articles/docker4seq.html#mirnaseq-workflow

3. Workflow for ChIP-seq analysis

For ChIP-seq data sets there were two main functions applied:

3.1 Genome indexing and mapping using BWA

```
bwaIndex(group=c("sudo","docker"), genome.folder=getwd(), genome.url=NULL, gtf.
url=NULL, dbsnp.file=NULL, g1000.file=NULL, mode=c("General","GATK","miRNA","nc
RNA"), mb.version=NULL, mb.species=NULL, rc.version=NULL, rc.species=NULL,
length=NULL)

bwa (group=c("sudo","docker"),fastq.folder=getwd(), scratch.folder="/data/scrat
ch", genome.folder, seq.type=c("se","pe"), threads=1, sample.id, circRNA=FALSE)
```

BWA indexing uses ENSEMBL genomic data. User has to provide the URL (uscs.urlgenome) for the file chromFa.tar.gz related to the organism of interest and the path to the folder where the index will be generated (genome.folder). The parameter gatk has to be set to FALSE if it is not required for ChIPseq genomic index creation.

The accepted input files are in .bam format, generated with any tool BWA. The user has to provide two condition treatment and background for peak calling procedure. The peak calling is performed on treatment using the background information to model the noise of the system and for this reason a background from same system is provided which is generated using a immunoglobulin independent treatment.

The parameters for peak calling is implemented in docker4seq package with the default setting as implemented in the baseSpace App. The user having experience with peak calling algorithm can tweak options for peak calling procedure.

3.2 Peak calling and annotation using MACS

```
system("wget 130.192.119.59/public/test.chipseqCounts.zip")
unzip("test.chipseqCounts.zip")
setwd("test.chipseqCounts")
library(docker4seq)
macs2(group=c("sudo","docker"), control.bam, chipseq.bam, experiment.name, histone.marks=FALSE, broad.cutoff=0.1, qvalue=0.05, organism=c("hs", "mm"))
```

User needs to create the following folders:

```
+ mock.folder, where the fastq.gz file for the control sample is located. For control sample we refer to ChIP with IgG only or input DNA.

+ test.folder, where the fastq.gz file for the ChIP of the sample to be analysed.

+ output.folder, where the R script embedding the above script is located.
```

There were certain limitations to the ChIP-seq workflow, in the BaseSpace App such as each input sample needed to be provided only as unique BAM file because the app did not allow the uploading of the fastq files. The BAM file provided needed to aligned with a external tool The app only supported peak-calling via background noise definition by external resources.

More details and step-by-step guidelines can be found at: https://kendomaniac.github.io/docker4seq/articles/docker4seq.html#chipseq-workflow

Validation of workflows in RBP

The workflows designed and described above for mRNA and miRNA data sets were successfully embedded in the docker4seq package which is a module in the RBP and were validated in different research articles.

a) Using mRNA data sets

The (Susanna Zucca, et. al, 2019), was a study performed to reveal the metabolism of the coding and long non-coding RNA role in the Amyotrophic Lateral Sclerosis (ALS) pathogenesis. The data was obtained from the Peripheral Blood Mononuclear Cells (PBMCs) from sporadic and mutated Amyotrophic Lateral Sclerosis patients with specific mutations in FUS, TARDBP, SOD1, VCP genes and healthy donors as controls. The aim of the study was to compare the coding and non-coding RNAs and to study the difference of the diseases state and healthy controls, for sporadic and mutated patients.

The dataset used in this study included samples from 15 sporadic ALS (sALS) patients, 9 ALS patients with mutation in classical ALS-genes, 3 patients were mutated in SOD1 gene (SOD1-m1, SOD1-m2 and SOD1-m3), 3 in FUS gene (FUS-m1, FUS-m2 and FUS-m3), 2 in TARDBP gene (TARDBP-m1 and TARDBP-m2) and one in VCP gene (VCP-m1), as well as 7 healthy individuals (controls).

Human PBMCs were isolated from ALS patients and healthy controls, RNA was extracted from them and an RNA-seq library was prepared using the Illumina TruSeq Stranded RNA Library Prerp. Fastq files were generated using bcl2fastq2 (Version 2.17.1.14), their quality was checked FastQC software v0.11.6 and Cutadapt was used for adapter trimming. STAR 2.6 was used for read mapping to reference human genome GRCh38 genome assembly and RSEM v1.3.0 was applied for gene expression quantification. DEseq2, as an R package was used to perform differential expression analysis for all the transcripts coding and non-coding RNAs.

b) Using miRNA data sets

My miRNA-seq workflow was used in one of the paper in which I am co-author: Barbara Pardini, et,al, 2018. This study highlights the abnormalities associated in Bladder Cancer (BC), which is one of the most regularly occuring malignancies worldwide. The BC screening and early primary diagnosis is believed to improve patient quality of life and their survival rate. The main highlight of the study is based on identifying the urinary miRNA profiles associated with the BC and different clinic-pathological subtypes by using NGS techniques. The most significant miRNAs were used to build model to predict the BC.

The dataset involved in this study included only men. There were 114 samples in total (from 66 BC cases and 48 controls) used in the analyses. RNA was extracted from the urine, stored and processed together with library preparation (small RNA-seq) using NEBNext Multiplex Small RNA Library Prep Set for Illumina. miRNA was quantified by qPCR. miRNA biomarkers were validated in independent urine samples using the miRCURY LNA. For the computational and statistical analyses, my developed pipeline was successfully used to analyse the associated miRNA-seq dataset. A total of 98 DE miRNAs were identified, out of which 5 miRNAs (miR-30a-5p, miR-205-5p, miR-584, let-7c and miR-7706) were associated with a

Predictive Power (PP) higher than 0.7 when performed using logistic regression analysis. (Barbara Pardini, et,al, 2018)

Other studies that have implemented my designed workflow, embedded in docker4seq for both mRNA and miRNA include: Ambrosio S. *et al.*, 2017; Ferrero G. *et al.*, 2017; Pardini B. *et al.*, 2018; Adamo A. *et al.*, 2019; Arcà B. *et al.*, 2019.

Discussion

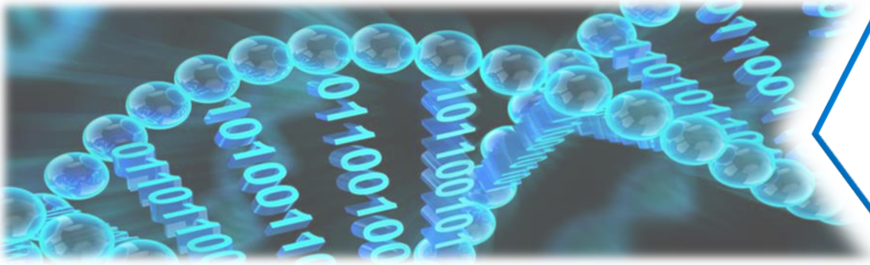
Reproducibility has been supported and used as a measure to establish reliability of the scientific literature and the knowledge we believe in. The increasing dependability on research data and bioinformatics and computational techniques for its capacity to reproduce results is holding an increasing importance to the scientific community. Aiming in case where a novel result is claimed, other researches should be able to reproduce them given that the raw data and knowledge of the experiment is provided. The reproducible analysis of high throughput sequencing data is a challenging and computationally intensive task, as it involves in depth understanding of not only methods and tools used but also data, infrastructure and software. Until now, reproducibility of workflows has been a non-trivial task. It is further complicated by the interchangeably used reproducibility terminologies (such as repeatability and replicability), myriad tools and heterogeneous platforms available to support workflow design and implementation.

The detailed literature review of previous studies shows that reproducing previously analysed results can be challenging due to insufficient and sometimes erroneous reporting such as technical errors and bias to publish positive results and information. Moreover, there exists a lack of clarity in terminologies and concepts associated with reproducibility in the literature. In this context, if we cannot agree on the concepts of reproducibility then we have little chance to reproduce existing results as described in detail in research publications. The result of this is that considerable effort is lost for reproducing experiments/analysis and lack of faith in research outcomes is manifest causing an additional burden of reproducibility crisis.

Important is that scientific claims made through experiments last the test of time as it’s very difficult to practically implement and pursue scientific discoveries if the foundations of such evidence are weak or if the reproducibility of claims is not possible. Therefore, essentially rethinking current research methods and consider standardization of research practices. This thesis aims to support reproducibility of bioinformatics workflows by identifying and characterizing implicit and explicit assumptions with intricate details associated with workflow design and implementation. These assumptions are often considered needless to be stated, but lead to various factors that ultimately impact on the reproducibility of workflows.

Reproducibility has been time and again shown to be the core principle for any scientific research is built upon theory and experiments, verified and extended ideally through open science. On the other hand, irreproducible researches create a wastage of resources, time and effort misleading the research community. The main issue is that the intricate information associated with modern workflow implementations suitable for large scale -omics data has reproducibility requirements that are not well understood. Understanding reproducibility requirements of bioinformatics workflows and providing a clear definition of reproducibility and the associated dimensions have motivated in this thesis.

Chapter 4



Computational analysis of miRNAs from human biofluids and surrogate tissues

Chapter 4

Computational analysis of miRNAs from human biofluids and surrogate tissues

4.1 Introduction

miRNAs are small non coding RNAs ~ 22 nucleotides in length. Most of them are transcribed from DNA into primary miRNAs (pri-miRNAs) and then processed into precursor miRNAs and mature miRNAs. The majority of miRNAs interact with the 3' untranslated regions (UTRs) of target mRNAs, leading to down regulation of gene expression. In addition, miRNAs interactions with other regions such as 5' UTR, coding sequence and gene promoter regions have also been reported. Other studies have also shown that miRNAs travel to and from different subcellular compartments, thus maintaining a continuous and tight control over the rates of translation and transcription. Therefore, miRNAs are very important regulatory elements for normal development in numerous biological processes and abnormal expression of miRNAs has been linked with many different human diseases. (Makarova J. A. *et al.*, 2016)

What is more, miRNAs secreted in the extracellular fluids have been widely studied and demonstrated to have potential as biomarkers for different diseases, as well as behaving as signalling molecules in mediating cell to cell communications. (Ferrero G. *et al.*, 2018) In contrast to the RNA species in the cellular region, the extracellular miRNAs are highly stable and can resist degradation at room temperature for up to 4 days. (O'Brien J. *et al.*, 2018).

Studies on these extracellular miRNAs were reported for biological fluids such as plasma, serum, cerebrospinal fluid, saliva, breast milk, urine, tears, colostrum, peritoneal fluid, bronchial lavage, seminal fluid, ovarian follicular fluid etc (Weber J. A. *et al.*, 2010).

Taking all these into account, the aim of the current chapter is to demonstrate that my computational workflow can be used to identify the abundance and distribution of both common and unique miRNAs in different biospecimens. More than that, differential analysis of the uniquely observed miRNAs were cross-validated against previously published studies which found the corresponding miRNAs as biomarkers for different diseases. All the results obtained from both wet-lab experiments and bioinformatics analysis based on my designed pipeline were published in (Ferrero G. *et al.*, 2018), for which I was a co-author.

4.2 Material and Methods

Dataset

The datasets used for the above-mentioned study Giulio Ferrero, et.al, 2018 were obtained from a total of 125 samples of plasma-derived exosomes, out of which 48 were urine samples, 31 cervical scrapes and 39 stools samples, all from healthy donors (controls).

Stool and plasma samples – In a hospital-based study for colorectal cancer negative subjects were recruited as controls, we obtained the stool and plasma samples donors who tested negative for the colorectal cancer. Naturally evacuated stool was collected in special tubes with RNA stabilizing solution and stored at -80°C until RNA extraction was performed.

Urine sample – Urine samples were collected from the donors in the morning and stored at 4°C until the process of centrifugation at 3000g for 10mins. The urine supernatants were then transferred to other tubes and stored at -80°C until use.

Cervical scrapes – The cervical scrape samples were obtained from the women who only tested negative for HPV. The samples were collected and stored in Specimen Transport Medium (STEM) at -80°C until RNA extraction.

All the following wet-lab experiments were performed by other colleagues in my group: RNA isolation, exosome isolation from plasma, RNA extraction and library preparation. I have run the computational analysis, focused on investigating expression levels and patterns of miRNAs of the comparable sizes in the aforementioned four biospecimens obtained from human biofluids and surrogate tissues for diagnostic and screening programs. The aim of the study was to explore new potential biomarker for disease classification and a clear overview of the commonly and uniquely expressed miRNAs in different human biospecimens.

RNA isolation

Total RNA was isolated from samples with kits suited for each type of specimen. Total RNA was used to prepare sequencing libraries, subsequently sequenced on an Illumina Nextseq 500. The resulting fastq files were analysed with the miRNA workflow, developed as a part of the Reproducible Bioinformatics Project (described in detail in Chapter 3).

Exosome isolation from plasma

The plasma samples were obtained from 5-8ml of blood centrifuged for 10mins at 1000 rpm following to which the plasma aliquots (about 200-300 µl) were stored about at -80°C until next use. ExoQuick exosome precipitation solution (System Biosciences, Mountain View, CA, USA) according to the manufacturer's instructions with minor modifications were used to isolate Exosomes from 200 µl of plasma. the plasma was mixed with 50.4 µl of ExoQuick solution and refrigerated at 4°C overnight (at least 12 h). The mixture was then further centrifuged at 1500 g for 30 min. The exosome pellet was dissolved in 200 µl of nuclease free water; RNA was extracted immediately from the solution.

RNA extraction and quality control

Using the miRNeasy plasma/serum mini kit (Qiagen) and the QiaCube extractor (Qiagen), total RNA from plasma exosomes was extracted. Using the Total RNA Purification Kit (Norgen Biotek Corp), RNA from stool was extracted. The RNA from the cervical scrape was extracted from the samples stored in the STM, using the miRCURY RNA isolation Kit. Using the MIQE guidelines, the RNA quality and quantity was verified. Using the Qubit 2.0 fluorometer, RNA concentrations were quantified.

Library preparation

Small RNA library preparation was performed with the NEBNext Multiplex Small RNA Library Prep Set for Illumina (New England BioLabs Inc., USA) (Figure 4.1)

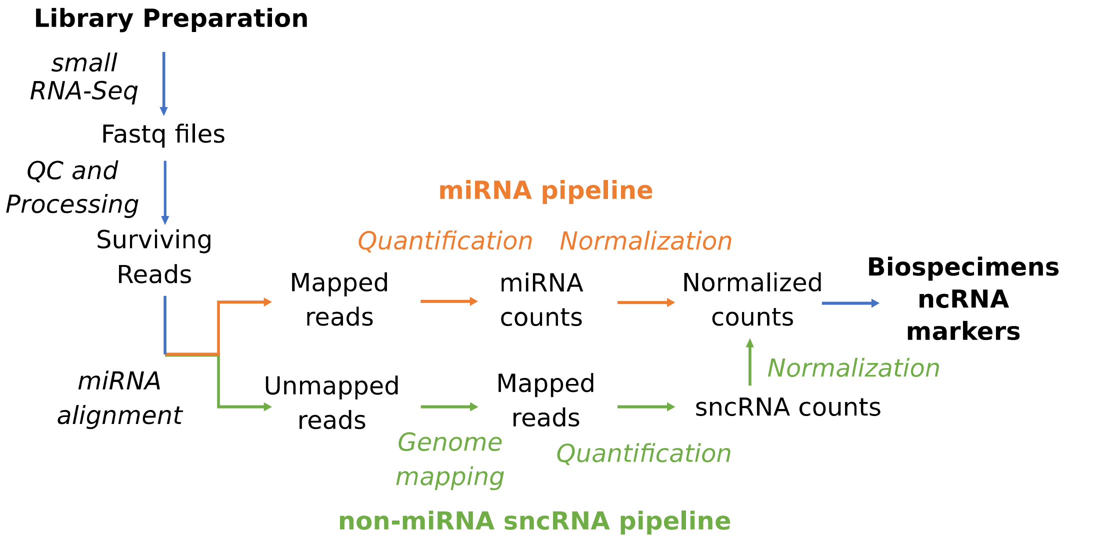


Figure 4.1: Schematic representation of library preparation and miRNA pipeline

Computational analysis of miRNAs

As described in the previous Chapter 3, miRNA fastq files were pre-processed and their quality was checked using FastQC, after which reads shorter than 14 nucleotides were filtered out/discarded. These were then adapter-trimmed using Cutadapt and subsequently mapped against the precursor miRNA sequences downloaded from miRbase (Release 21), using Shrimp algorithm. A count matrix made of integer values referring to the level of expression of mature miRNAs were generated using Bioconductor Genomic Ranges (GRanges) based script, which counts reads located in the region of miRNA precursors for which mature miRNAs are expected. This count matrix is then used for further analysis of differentially expressed genes.

The expression levels of miRNAs were identified in all different specimens and were then compared using Venn diagram and heatmap2 R function. PCA analysis was performed using prcomp R function and autoplot function from ggfortify R package. A list of validated miRNAs was annotated in miRWalk 2.0 database (Dweep H. and Gretz N., 2015) and then EnrichR (Chen E. Y. *et al.*, 2013), a web tool (https://amp.pharm.mssm.edu/Enrichr/) used for the functional enrichment analysis of miRNAs (Dweep H. and Gretz N., 2015). The uniquely identified and differentially expressed miRNAs were then cross-checked with numerous miRNAs already reported as biomarkers in tissues in altered state of disease.

4.3 Results

The mapping analysis of miRNA-seq data recorded remarkable read alignment rates (figure 27), with the highest number of uniquely-mapped reads found in urine samples, with a median read 12.38 million. Plasma exosomes had second highest median read of 11.34 million, while the median rates for stool samples and cervical scrapes were 4.88 million and 4.13 million, respectively (Figure 4.2)

A total of 1823 miRNAs were successfully identified and annotated using miRbase database, with 73.8% corresponding to plasma exosomes and 19.9% to cervical scrapes. The differential expression analysis embedded in my workflow also helped identify the miRNAs with the highest expression in each of the different biological tissue types - miR-486-5p in plasma exosomes, miR-320a in cervical scrapes, miR-6813-5p in stool samples, and miR-30a-5p in urines samples. (Figure 4.3B)

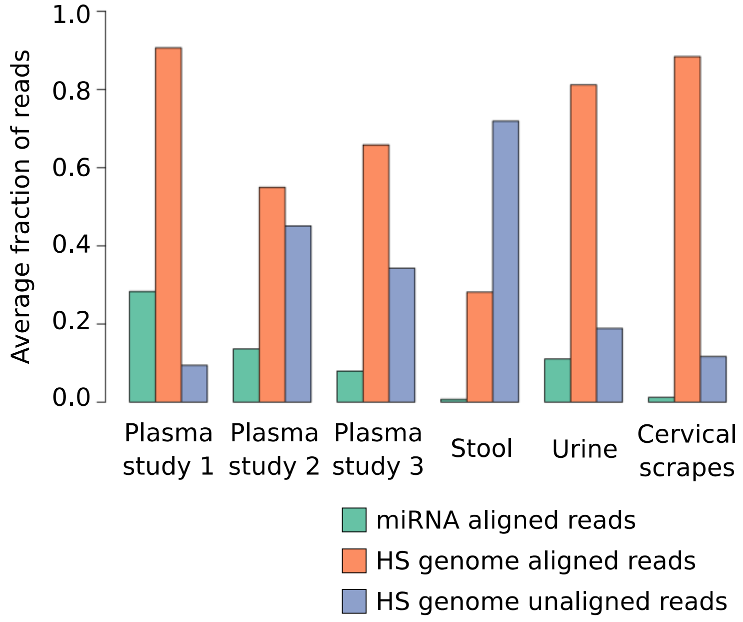


Figure 4.2: Barblot showing the fraction of reads aligned for different biospecimens

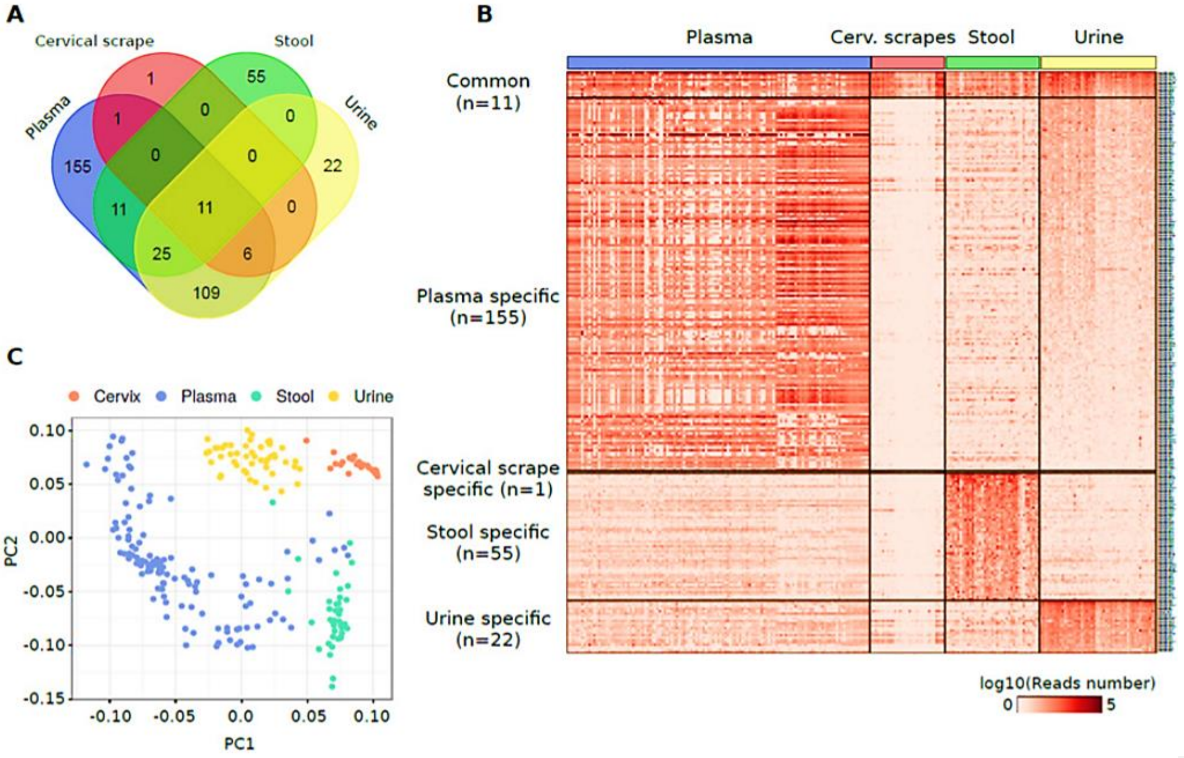


Figure 4.3: (**A**) Venn diagram reporting the number of miRNAs detected in different specimens from healthy individuals and their overlap. (**B**) Heat map showing the log10 number of normalized reads supporting the miRNAs specifically detected in one specimen or commonly detected among them. (**C**) PCA plot showing the small RNA-Seq datasets separation obtained using miRNAs detected in samples

miRNAs were commonly and uniquely identified in different biospecimens involved in the study. PCA visualization of expression results showed different miRNA clusters for each biospecimen. A pairwise correlation analysis provided comparable results. Out of these, 11 miRNAs were identified as common between all specimen types: miR-320a, miR-589-5p, miR-636, miR-1273a, miR-3960, miR-4419a, miR-4497, miR-4709-5p, miR-4792, miR-7641-1, and miR-7641-2 (Figure 28a). In contrast, 155 miRNAs were uniquely present in the plasma exosomes, comparably more than in any of the other biospecimen – stool samples had 55 miRNAs uniquely present, urine samples had 22 miRNAs and cervical scrapes had only 1 miRNA uniquely present. (Figure 28A).

In addition, PCA analysis of highly expressed miRNAs provided a good accuracy in the classification of different biospecimen (Figure 28C). Although not implemented in the Docker4Seq mRNA workflow, random forest classification was subsequently used and an accuracy percentage of 99.9% was obtained, with only one sample being misclassified. Moreover, all the miRNAs were analysed based on a discriminate capacity, using chi square statistic for attribute selection analysis. Out of these, three miRNAs (miR-204-5p, miR-5698, and miR-335-3p) recorded the highest merit scores/lowest p-values.

For a number of patients, multiple samples were collected from the same individual, such as both plasma and stool, or plasma and urine. The availability of such paired data for sets of 2 different biospecimens enabled a comparison analysis between the expression levels of miRNAs for the same patient. A low co-expression was observed between both plasma-stool or plasma-urine samples. However, there was one exception – miR-3665, which was observed to be a positive correlation (r=0.59, p=2.0*10-5) between plasma and urine sample (Figure 4.4).

An analysis to predict the miRNA isoforms (isomiRs) was performed. 832 isomiRs were detected in at least one specimen with more than20 supporting reads. There 94.4% of isomiRs found in plasma exosomes or urine sample consistently with consistently higher reads in these samples. IsomiRs with a 3’ variant of miR-486/miR-486-2 in plasma sample 5′ variant of miR-934 in urine sample, a 5′ variant of miR-7704 in cervical scrapes, and a 3′ variant of miR-583 in stool samples were isomiRs with higest number of supporting reads. Out of 11 common miRNAs that were previously identified, 8 were associated to an isomiR.

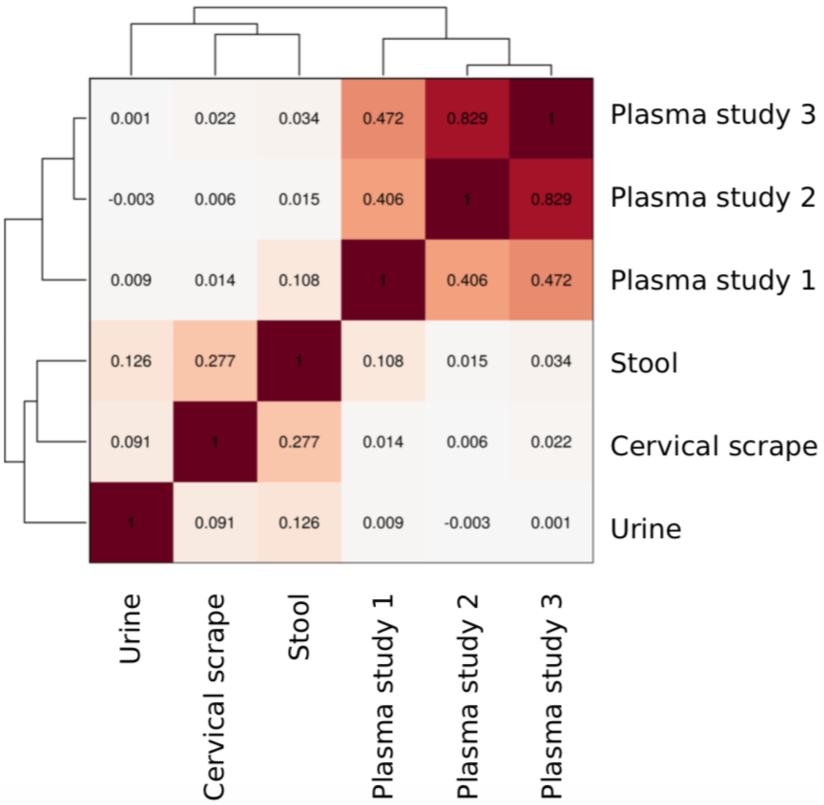


Figure 4.4: Correlation analysis of paired datasets.

The most commonly obtained miRNA (miR-320a) and its downregulation was shown to be associated with different disease including cancer (Xie F. *et al.*, 2017). miRNA (miR-589-5p) was identified in our study which is a good inhibitor of MAP3K8 and is associated with hepatocellular carcinoma which a suppressor of CD90+ cancer stem cells (Zhang Xi *et al.*, 2016). Another regulatory element, miR-636 also identified in the study was reported as a good biomarker for many diseases, such as diabetics and kidney diseases (Eissa S. *et al.*, 2016), pancreatic cancer (Schultz N. A. *et al.*, 2014), colorectal cancer (Slattery M. L. *et al.*, 2016). miR-4792, also identified using the current workflow was previously demonstrated to be dysregulated in nasopharyngeal carcinoma tissues (Li Y. and Chen X., 2015) and oral submucous fibrosis (Chickooree D. *et al.*, 2016).

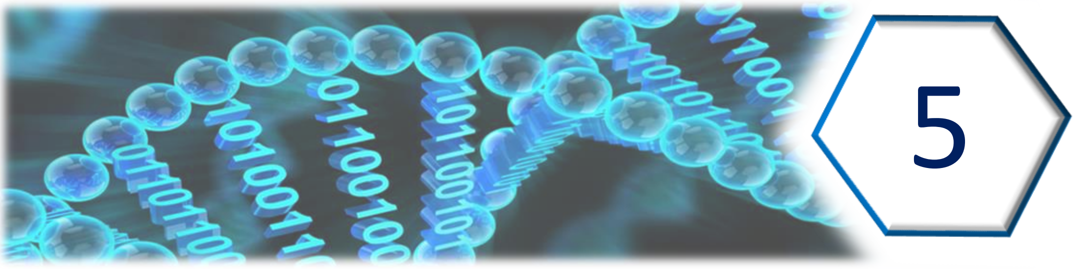
Interestingly, the rest of the commonly obtained miRNAs were never studied in detail. Although these miRNAs are found to be dysregulated in different diseases found in different biospecimens can be used as multispecies markers. Several studies also obtained comparable results on the available dataset from same anatomically related tissue. (Ben-Dov I. Z. *et al.*, 2016; Seashols-Williams S. *et al.*, 2016; Yeri A. *et al.*, 2017). Similarly, using our docker3seq embedded pipeline, miR-320a and miR-589-5p were observed to be highly expressed in the tested data sets and they were also found in all other observed datasets.

Discussion

The results described above demonstrated that my designed miRNA workflow (also embedded in the docker4seq package) was successful in identifying the miRNAs that were commonly and uniquely expressed in the four different biospecimens included in this study. In total, there were 400 miRNAs detected in one or more specimen types, while a large set of miRNAs were expressed only in plasma exosomes, but only a few miRNAs were specific to stool or urine, and just one was found to be specific to cervical scrapes. However, there were 109 commonly expressed and shared miRNAs between plasma exosomes and urine.

More than that, considering the total number of highly expressed miRNAs, these were successfully and efficiently classified and separated by their original biological type. This classification is very important in biomarker studies, which could be representing an altered state of tissue type in its association with various diseases. In contrast, I identified 11 miRNAs with similar expression patterns in all four specimens. Overall, the data studied provided an insight to the human miRNome for different biospecimens of healthy individuals.

Chapter 5



Conclusions and Future Work

Chapter 5

Conclusions and Future Work

Reproducibility has always been one of the most important aspects for high quality scientific research, but it has also posed major issues that continue to be addressed. Moreover, the increase in demand and use of bioinformatics approaches has led to more systematic strategies for solving reproducibility problems. One of these is Reproducible Bioinformatics Project (RBP), developed with the aim of reducing the reproducibility issues, since it is a framework helping users to collate, analyse and visualize data. The workflows implemented in RBP are designed to bring back in the hands of life scientists the basic bioinformatics required for the majority of transcriptomics and genomics studies, e.g. going from fastq to differential expression or calling ChIPseq peaks. For this reason, a GUI is provided together with workflows vignette and a you tube channel describing how to run the workflows using the GUI.

At the same time RBP is granting the robustness and reproducibility of the workflows being designed on the basis of the reproducibility rules proposed by Sandve and assembled using docker containerization which guarantees computational reproducibility.

5.1 RBP workflows for mRNAseq, miRNAseq and ChIPseq analysis

The stable version of the current RBP has three functional workflows for the analysis of mRNAseq, miRNAseq and ChIPseq. These enable the identification and study of expression patterns of different genetic elements in a tissue specific manner. A test dataset is also provided for beginners to test and familiarize themselves with each workflow.

mRNAseq pipeline offers an insight into the transcriptomic state of diseases and is designed detect differential expressed genes and transcripts. **miRNAseq** workflow analyses mature miRNAs, identifies unannotated 5P or 3P mature mRNAs, looks at their distribution and abundance, and quantifies their expression levels. The third workflow in RBP is **ChIPseq**, which helps the users to examine the mechanism of transcription factors (TF) and chromatin-associated proteins, to determine their interaction with DNA and to understand its potential for regulating gene expression.

5.2 Computational analysis of miRNAs from human biospecimens

Diagnostic and the therapeutic procedures demand for less invasive techniques to analyse different biospecimens for the study biomarker detection in complex diseases. This emphasizes the importance of our computational workflows, which can be used to analyse various sequencing data for both predicting the presence of specific regulatory elements and understanding their pathogenesis in relation to different diseases. The pipeline developed for analysis of miRNAseq data was used to evaluate the miRNA profiles in different tissues and human biofluids obtained from healthy individuals through non-invasive methods. The analysis was performed on plasma exosomes, urine, stool and cervical scrap samples and investigated the presence, distribution and expression of miRNAs in each biological condition.

A large set of uniquely-present miRNAs (n=155) was found in plasma exosomes, while in stool and urine samples there were less miRNA uniquely identified, and only one miRNA was associated with the cervical scraps. However, plasma exosomes and urine samples shared approximately 100 common miRNAs, and even fewer were recorded between the other biospecimens. The four biological tissues tested were also classified based on the highly expressed miRNAs, thus proving the capability of our workflow to be used for predicting and identifying possible biomarkers. For example, the most commonly obtained miRNA, miR-320a was cross-checked with various published studies and was found to be downregulated in conditions like cancer. Similarly, miR-589-5p was reported as a good inhibitor of MAP3K8 and associated with hepatocellular carcinoma. Other miRNAs identified were shown as potential biomarkers involved in diseases such as diabetes and kidney disease, pancreatic and colorectal cancer (miR-636), and nasopharyngeal carcinoma (miR-4792).

In conclusion, the 3 workflows for mRNAseq, miRNAseq and ChIPseq analysis represent our first successful attempt to build an RBP platform supporting scientific reproducibility. They are wrapped in a docker images and supported by video tutorials which describe the use of each workflow in a step-by-step manner. Workflows for mRNAseq and miRNAseq analysis have also been **cross-validated** in a number of studies published in peer-reviewed journals.

More than that, the high number of online visitors and users demonstrate the demand of our tool and its increasing popularity (Figures 5.1 – 5.4). This is because the developed pipelines embedded into a docker4seq platform confer **flexibility** for optimization and further development of individual tasks for each container and it uses a **user-friendly interface**. Other future workflows are also under development for the analysis of variants calling in patient derived xenografts (PDX) from RNAseq and EXOMEseq data, for single cell analysis and metagenomics.

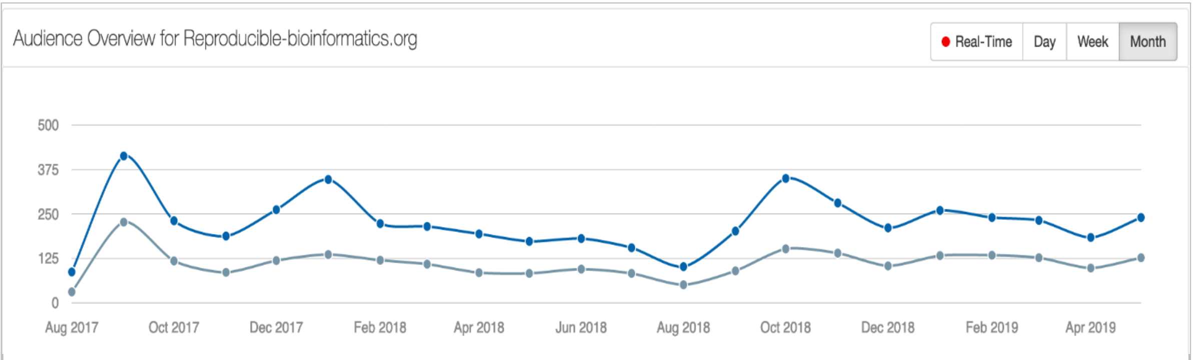


Figure 5.1: Plot representation overview of audience of RBP

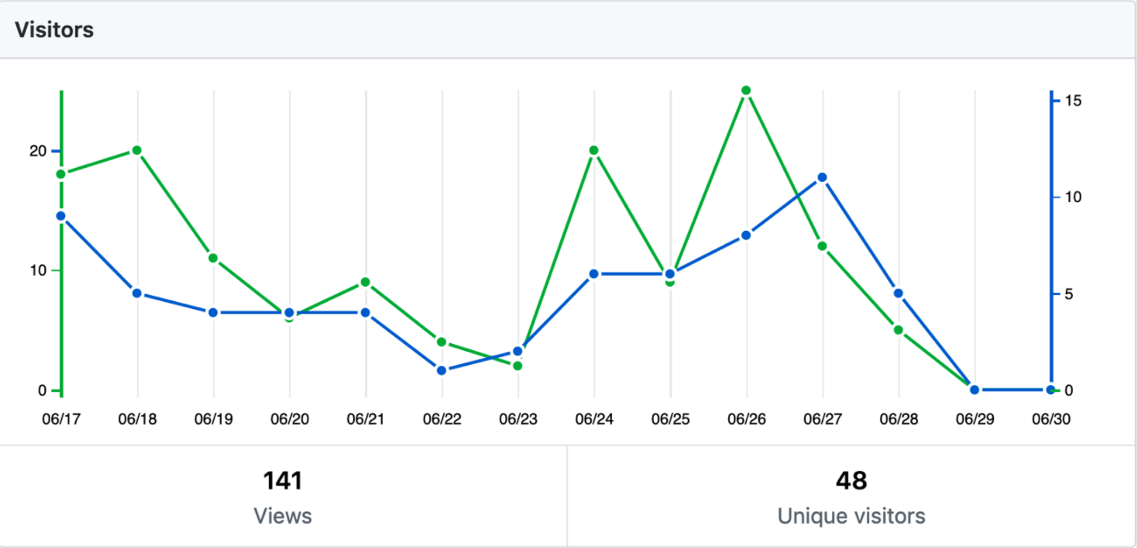
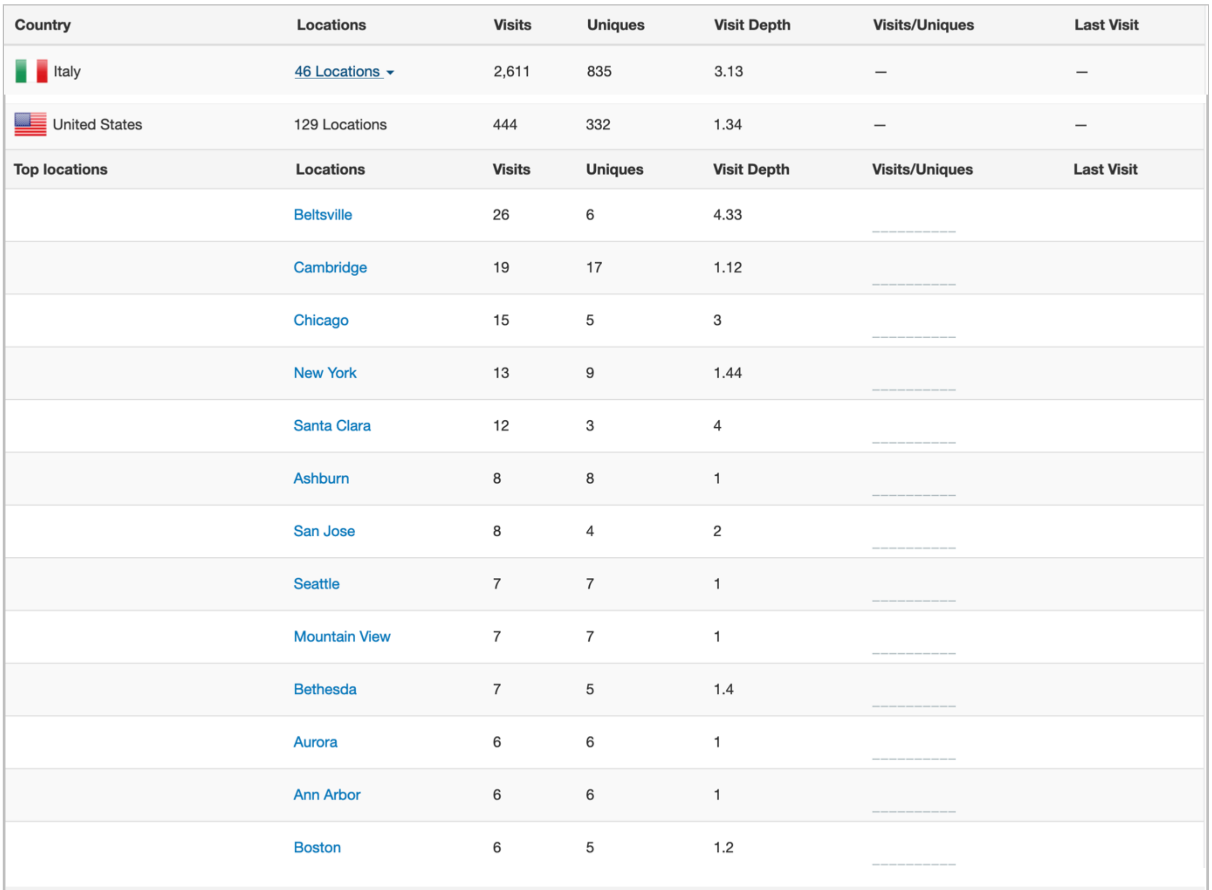
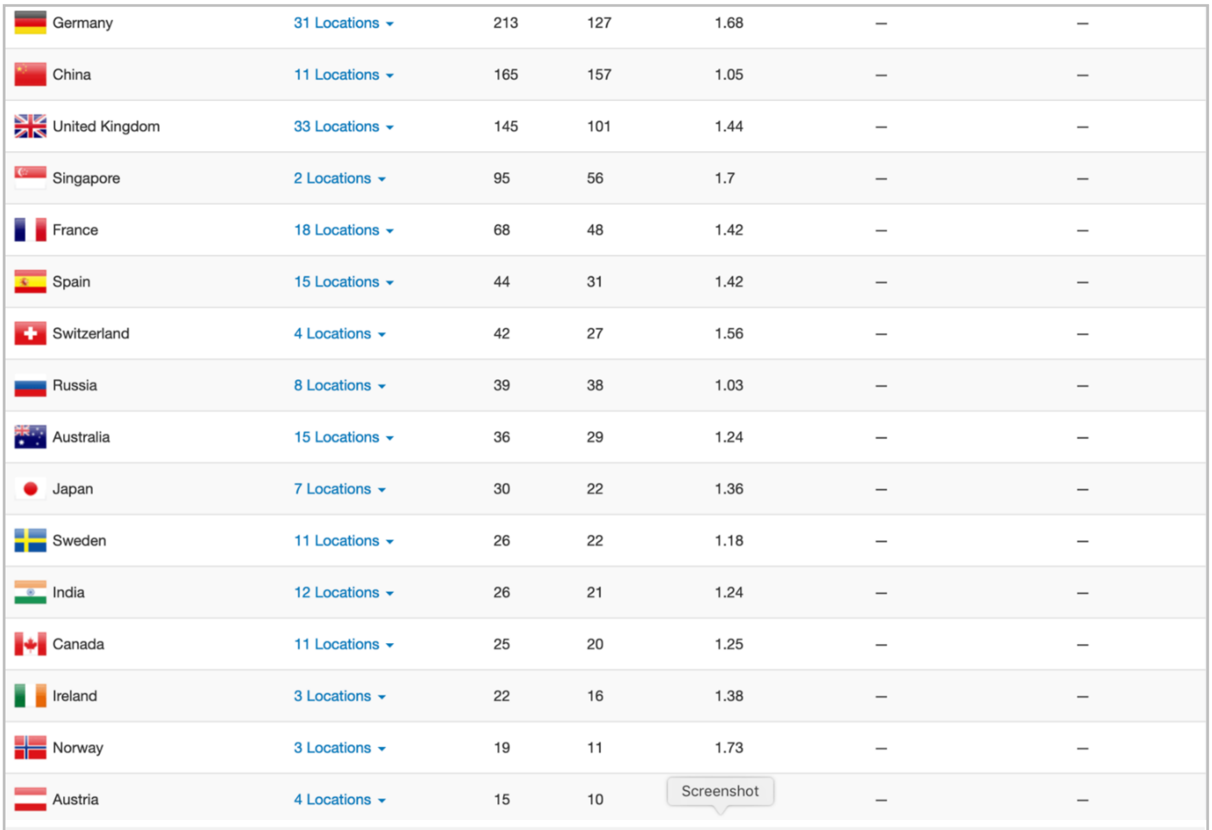


Figure 5.2: Plot representation of visitors to ourGitHub link



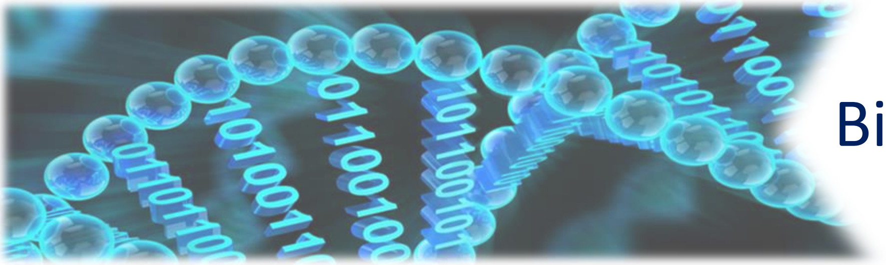
Country	Locations	Visits	Uniques	Visit Depth	Visits/Uniques	Last Visit
Italy	46 Locations ▾	2,611	835	3.13	–	–
United States	129 Locations	444	332	1.34	–	–
Top locations	Locations	Visits	Uniques	Visit Depth	Visits/Uniques	Last Visit
	Beltsville	26	6	4.33		
	Cambridge	19	17	1.12		
	Chicago	15	5	3		
	New York	13	9	1.44		
	Santa Clara	12	3	4		
	Ashburn	8	8	1		
	San Jose	8	4	2		
	Seattle	7	7	1		
	Mountain View	7	7	1		
	Bethesda	7	5	1.4		
	Aurora	6	6	1		
	Ann Arbor	6	6	1		
	Boston	6	5	1.2		

Figure 5.3: RBP Visitors from United States and Italy



Germany	31 Locations ▾	213	127	1.68	–	–
China	11 Locations ▾	165	157	1.05	–	–
United Kingdom	33 Locations ▾	145	101	1.44	–	–
Singapore	2 Locations ▾	95	56	1.7	–	–
France	18 Locations ▾	68	48	1.42	–	–
Spain	15 Locations ▾	44	31	1.42	–	–
Switzerland	4 Locations ▾	42	27	1.56	–	–
Russia	8 Locations ▾	39	38	1.03	–	–
Australia	15 Locations ▾	36	29	1.24	–	–
Japan	7 Locations ▾	30	22	1.36	–	–
Sweden	11 Locations ▾	26	22	1.18	–	–
India	12 Locations ▾	26	21	1.24	–	–
Canada	11 Locations ▾	25	20	1.25	–	–
Ireland	3 Locations ▾	22	16	1.38	–	–
Norway	3 Locations ▾	19	11	1.73	–	–
Austria	4 Locations ▾	15	10	Screenshot	–	–

Figure 5.4: RBP Visitors from rest of the world



Bibliography

Bibliography

Adamo A. *et al.* (2019). "Extracellular Vesicles Mediate Mesenchymal Stromal Cell-Dependent Regulation of B Cell PI3K-AKT Signaling Pathway and Actin Cytoskeleton". Frontiers in Immunology 10: 446

Altintas I. *et al.* (2004). "Kepler:An Extensible System for Design and Execution of Scientific Workflows.", IEEE Computer Society

Ambrosio S. *et al.* (2017). "LSD1 mediates MYCN control of epithelial-mesenchymal transition through silencing of metastatic suppressor NDRG1 gene." Oncotarget 8(3): 3854-3869

Arca B. *et al.* (2019). "MicroRNAs from saliva of anopheline mosquitoes mimic human endogenous miRNAs and may contribute to vector-host-pathogen interactions." Scientific Reports 9(1): 2955.

Arigoni M. *et al.* (2013). "miR-135b coordinates progression of ErbB2-driven mammary carcinomas through suppression of MID1 and MTCH2". American Journal of Pathology 182(6): 2058-2070.

Baker M. et al. (2016). "Is there a reproducibility crisis? - A Nature survey lifts the lid on how researchers view the 'crisis' rocking science and what they think will help". Nature Publishing Group 533, Issue 7604

Babak B. R. *et al.* (2017). "An Introduction to Docker and Analysis of its Performance". IJCSNS International Journal of Computer Science and Network Security 17(3)

Baddal B. *et al.* (2015). "Dual RNA-seq of Nontypeable Haemophilus influenzae and Host Cell Transcriptomes Reveals Novel Insights into Host-Pathogen Cross Talk". mBio 6(6): e01765-01715

Baggerly K. A. and Coombes K. R. (2009). "Deriving chemosensitivity from cell lines: Forensic bioinformatics and reproducible research in high-throughput biology". The Annals of Applied Statistics 3(4): 1309-1334

Barba L. A. (2018). “Terminologies for Reproducible Research". Preprint arXiv: 1802.03311

Barzago C. *et al.*, (2016). “Commentary: A novel infection- and inflammation-associated molecular signature in peripheral blood of myasthenia gravis patients”. Journal of Neurology & Neuromedicine 1(7): 24-27

Begley C. G. and Ioannidis J. P. (2015). "Reproducibility in science: improving the standard for basic and preclinical research". Circulation Research 116(1): 116-126

Bello N. M. *et al.* (2018). "Invited review: Reproducible research from noisy data: Revisiting key statistical principles for the animal sciences" Journal of Dairy Science 101(7): 5679-5701

Ben-Dov I. Z. *et al.* (2016). “Cell and Microvesicle Urine microRNA Deep Sequencing Profiles from Healthy Individuals: Observations with Potential Impact on Biomarker Studies”. PLoS ONE 11(1): e0147249. doi:10.1371/journal.pone.0147249

Benjamin L. *et al.* (2014). “Stem Cell Transcriptional Networks Methods and Protocols”. Springer Science

Bland J. M., *et al.* (2010). "Statistical methods for assessing agreement between two methods of clinical measurement". International Journal of Nursing Studies 47(8): 931-936

Blatecky A. (2017). “Reproducibility: A primer on semantics and implications for research”. RTI Press Publications: 978-1-934831-21-2

Bo Li and Dewey C. N. (2011). “RSEM: accurate transcript quantification from RNA-Seq data with or without a reference”. BMC Bioinformatics, 12:323

Boettiger C. (2015). “An introduction to Docker for reproducible research, with examples from the R environment”. ACM SIGOPS Operating Systems Review, Special Issue on Repeatability and Sharing of Experimental Artifacts

Buffalo V. (2015). “Bioinformatics Data Skills: Reproducible and Robust Research with Open Source Tools”, Edition 1:1-538

Buza T. M. *et al.* (2019). "iMAP: an integrated bioinformatics and visualization pipeline for microbiome data analysis". BMC Bioinformatics 20(1): 374

Califano A. and Rigoutsos I. (1993). “Flash: a fast look-up algorithm for string homology”. Computer Vision and Pattern Recognition, 1993 IEEE Computer Society Conference on: 353–359

Chen E. Y. *et al.* (2013). “Enrichr: interactive and collaborative HTML5 gene list enrichment analysis tool”. BMC Bioinformatics 14:128

Chickooree D. *et al.* (2016). “A preliminary microarray assay of the miRNA expression signatures in buccal mucosa of oral submucous fibrosis patients”. Oral Pathology & Medicine 45(9)

Chirigati F. and Freire J. (2017). “Provenance and Reproducibility”. Encyclopedia of Database Systems: 1-5

Cohen-Boulakia, S., *et al.* (2017). "Scientific workflows for computational reproducibility in the life sciences: Status, challenges and opportunities". Future Generation Computer Systems 75: 284-298

Conesa A. *et al.* (2016). "A survey of best practices for RNA-seq data analysis." Genome Biology, 17:13

Cordero F. *et al.* (2012). "Optimizing a massive parallel sequencing workflow for quantitative miRNA expression analysis." PLoS One 7(2): e31630

Cribbs A. P. *et al.* (2019). “CGAT-core: a python framework for building scalable, reproducible computational biology workflows”. F1000Research

Dobin A. *et al.* (2013). "STAR: ultrafast universal RNA-seq aligner." Bioinformatics 29(1): 15-21

Drummond C. (2009). “Replicability is not Reproducibility: Nor is it Good Science“. Evaluation Methods for Machine Learning

Dweep H. and Gretz N. (2015). “miRWalk2.0: a comprehensive atlas of microRNA-target interactions”. Nature Methods 12(8): 697-697

Eissa S. *et al.* (2016). “Urinary exosomal microRNA panel unravels novel biomarkers for diagnosis of type 2 diabetic kidney disease”. Journal of Diabetes and its Complications 30(08): 1585-1592

Emde A. K. *et al.* (2010). "MicroRazerS: rapid alignment of small RNA reads". Bioinformatics 26(1): 123-124

Ewels P. *et al.* (2016). “MultiQC: summarize analysis results for multiple tools and samples in a single report”. Bioinformatics 32(19): 3047–3048

Fanelli D. *et al.* (2010). "Positive" results increase down the Hierarchy of the Sciences". PLoS One 5(4): e10068

Felekkis K. *et al.* (2010). “microRNAs: a newly described class of encoded molecules that play a role in health and disease”. Hippokratia 14 (4): 236-240

Feng J. *et al.* (2012). "Identifying ChIP-seq enrichment using MACS". Nature Protocols 7(9): 1728-1740

Ferrero G. *et al.* (2018). “Small non-coding RNA profiling in human biofluids and surrogate tissues from healthy individuals: description of the diverse and most represented species”. Oncotargets 9(3): 3097-3111

Fernandes A. D. *et al.* (2013). "ANOVA-like differential expression (ALDEx) analysis for mixed population RNA-Seq". PLoS One 8(7): e67019

Fisch K. M. *et al.* (2015). "Omics Pipe: a community-based framework for reproducible multi-omics data analysis". Bioinformatics, 31(11): 1724-1728

Freire J. *et al.* (2012). “Making Computations and Publications Reproducible with VisTrails”. Computing in Science and engineering

Gil Y. et al. (2011). “WINGS: Intelligent Workflow-Based Design of Computational Experiments”. IEEE Intelligent Systems 26(1)

Goecks J. *et al.* (2010). “Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences”. Genome Biology 11: R86

Golosova O. *et al.* (2014). "Unipro UGENE NGS pipelines and components for variant calling, RNA-seq and ChIP-seq data analyses". PeerJ 2: e644

Griffiths-Jones S. *et al.* (2006). "miRBase: microRNA sequences, targets and gene nomenclature". Nucleic Acids Research 34(Database issue): D140-144

Guimera R. V. (2011). “Bcbio-nextgen: Automated, distributed, next-gen sequencing pipeline”. EMBnet.journal. 17(30)

Hackenberg M. *et al.* (2009). "miRanalyzer: a microRNA detection and analysis tool for next-generation sequencing experiments". Nucleic Acids Research 37(Web Server issue): W68-76

Jiang H. *et al.* (2014). “Skewer: a fast and accurate adapter trimmer for next-generation sequencing paired-end reads”. BMC Bioinformatics 15:182

Köster J. and Rahmann S. (2012). “Snakemake--a scalable bioinformatics workflow engine”. Bionformatics 28(19):2520-2

Leipzig, J. (2017). "A review of bioinformatic pipeline frameworks". Briefings in Bioinformatics 18(3): 530-536

Li H. and Durbin R. (2009). "Fast and accurate short read alignment with Burrows-Wheeler transform". Bioinformatics 25(14): 1754-1760

Li J. *et al.* (2014). "Bioinformatics pipelines for targeted resequencing and whole-exome sequencing of human and mouse genomes: a virtual appliance approach for instant deployment". PLoS One 9(4): e95217

Li Y. and Chen X. (2015). “miR-4792 inhibits epithelial–mesenchymal transition and invasion in nasopharyngeal carcinoma by targeting FOXC1”. Biochemical and Biophysical Research Communications 468(4): 863-869

Love M. I. *et al.* (2014). “Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2”. Genome Biology 15: 550

Lowe R. *et al.* (2017). "Transcriptomics technologies". PLoS Computational Biology 13(5): e1005457

Makarova J. A. *et al.* (2016). "Intracellular and extracellular microRNA: An update on localization and biological role". Progress in Histochemistry and Cytochemistry 51(3-4):33-49

Martin M. (2010). “Cutadapt removes adapter from sequence from high-throughput sequecning reads”. EMBLnet.journal

McArthur S. L. (2019). "Repeatability, Reproducibility, and Replicability: Tackling the 3R challenge in biointerface science and engineering". Biointerphases 14(2): 020201

Nekrutenko A. and Taylor J. (2012). "Next-generation sequencing data interpretation: enhancing reproducibility and accessibility". Nature Reviews Genetics 13(9): 667-672

O'Brien J. *et al.* (2018). "Overview of MicroRNA Biogenesis, Mechanisms of Actions, and Circulation". Frontiers in Endocrinology 9: 402

Pardini B. *et al.* (2018). “microRNA profiles in urine by next-generation sequencing can stratify bladder cancer subtypes”. Oncotargets 9(29): 20658-20669

Pardini B. *et al.* (2018). "MicroRNAs as markers of progression in cervical cancer: a systematic review". BMC Cancer 18(1): 696

Patro R. *et al.* (2017). "Salmon provides fast and bias-aware quantification of transcript expression". Nature Methods 14(4): 417-419

Pellizzari, E., *et al.* (2017). “Reproducibility: A Primer on Semantics and Implications for Research”. RTI Press Book series

Pepke S. *et al.* (2009). “Computation for ChIP-seq and RNA-seq studies”. Nature Methods 6: S22–S32

Piccolo S. R. and Frampton M. B. (2016). "Tools and techniques for computational reproducibility". Gigascience 5(1): 30

Popper K. et al. (2002). “The Logic of Scientific Discovery”. Edition 2: 1-544

Preeyanon L. et al. (2016). “Reproducible Bioinformatics Research for Biologists”. Brown Chapter

Prinz F. *et al.* (2011). "Believe it or not: how much can we rely on published data on potential drug targets?". Nature Reviews in Drug Discovery 10(9): 712

Rasmussen R. K. *et al.* (2006). “Efficient q-Gram Filters for Finding All ε-Matches over a Given Length”. Journal of Computational Biology 13(2)

Ronen R. *et al.* (2010). "miRNAkey: a software for microRNA deep sequencing analysis". Bioinformatics 26(20): 2615-2616

Rumble S. M. *et al.* (2009). "SHRiMP: accurate mapping of short color-space reads." PLoS Computational Biology 5(5): e1000386

Sadedin, S. P. *et al.* (2015). "Cpipe: a shared variant detection pipeline designed for diagnostic settings". Genome Medicine 7(1): 68

Sandve G. K. *et al.* (2013). "Ten simple rules for reproducible computational research". PLoS Computational Biology 9(10): e1003285

Schaffer S. *et al.* (1985). “Leviathan and the air-pump”. Princeton University Press

Schloss P. D. *et al.* (2018). "Identifying and Overcoming Threats to Reproducibility, Replicability, Robustness, and Generalizability in Microbiome Research". mBio 9(3)

Schultz N. A. *et al.* (2014). “MicroRNA Biomarkers in Whole Blood for Detection of Pancreatic Cancer”. JAMA 311(4):392–404

Seashols-Williams S. *et al.* (2016). “High-throughput miRNA sequencing and identification of biomarkers for forensically relevant biological fluids”. Electrophoresis 37(21)

Singer J. *et al.* (2018). "NGS-pipe: a flexible, easily extendable and highly configurable framework for NGS analysis". Bioinformatics 34(1): 107-108

Slattery M. L. *et al.* (2016). “Colorectal tumor molecular phenotype and miRNA: expression profiles and prognosis”. Modern Pathology 29: 915–927

Soneson C. and Delorenzi M. (2013). “A comparison of methods for differential expression analysis of RNA-seq data”. BMC Bioinformatics 14:91

Srivastava A. *et al.* (2016). "RapMap: a rapid, sensitive and accurate tool for mapping RNA-seq reads to transcriptomes". Bioinformatics 32(12): i192-i200

Teng M. *et al.* (2016). “A benchmark for RNA-seq quantification pipelines”. Genome Biology 17(74)

Wang W. C. *et al.* (2009). "miRExpress: analyzing high-throughput sequencing data for profiling microRNA expression". BMC Bioinformatics 10: 328

Warr W. A. *et al.* (2012). "Scientific workflow systems: Pipeline Pilot and KNIME". Journal of Computer-Aided Molecular Design 26(7): 801-804

Weber J. A. *et al.* (2010). "The microRNA spectrum in 12 body fluids". Clinical Chemistry 56(11): 1733-1741

Wingett S. W. *et al.* (2018). "FastQ Screen: A tool for multi-genome mapping and quality control". F1000Res 7: 1338

Wolstencroft K. *et al.* (2013). "The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud". Nucleic Acids Research 41(Web Server issue): W557-561

Xie F. *et al.* (2017). “miRNA-320a inhibits tumor proliferation and invasion by targeting c-Myc in human hepatocellular carcinoma”. OncoTargets and Therapy 10: 885-894

Yeri A. *et al.* (2017). “Total Extracellular Small RNA Profiles from Plasma, Saliva, and Urine of Healthy Subjects”. Scientific Reports 7(44061)

Zhang C. *et al.* (2017). "Evaluation and comparison of computational tools for RNA-seq isoform quantification". BMC Genomics 18(1): 583

Zhang Y. *et al.* (2008). "Model-based analysis of ChIP-Seq (MACS)." Genome Biol 9(9): R137

Zucca S. *et al.* (2019). "RNA-Seq profiling in peripheral blood mononuclear cells of amyotrophic lateral sclerosis patients and controls". Sci Data 6: 190006

Zhang X. *et al.* (2016). “miR-589-5p inhibits MAP3K8 and suppresses CD90+ cancer stem cells in hepatocellular carcinoma”. Journal of Experimental & Clinical Cancer Research 35(17)

Zyla J. *et al.* (2019). "Gene set enrichment for reproducible science: comparison of CERNO and eight other algorithms". Bioinformatics, btz447

Online sources:

Bradnam K. (2015). BioDocker and BioBoxes: the containerization of bioinformatics – http://www.acgt.me/blog/2015/8/25/biodocker-and-bioboxes-the-containerization-of-bioinformatics

Fernando A. (2018). Why it’s Important to Keep Your Containers Small and Simple – https://hackernoon.com/why-its-important-to-keep-your-containers-small-and-simple-618ced7343a5

MacKenzie R. J. (2019). Repeatability vs. Reproducibility – https://www.technologynetworks.com/informatics/articles/repeatability-vs-reproducibility-317157

Rehman J. (2013). Cancer research in crisis: Are the drugs we count on based on bad science? – http://www.salon.com/2013/09/01/is_cancer_research_facing_a_crisis/

Vaughan-Nichols S. J. (2018). What is Docker and why is it so darn popular? – https://www.zdnet.com/article/what-is-docker-and-why-is-it-so-darn-popular/

Beginner Pages and Tutorials:

bcl2fastq - https://support.illumina.com/content/dam/illumina-support/documents/documentation/software_documentation/bcl2fastq/bcl2fastq_letterbooklet_15038058brpmi.pdf

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

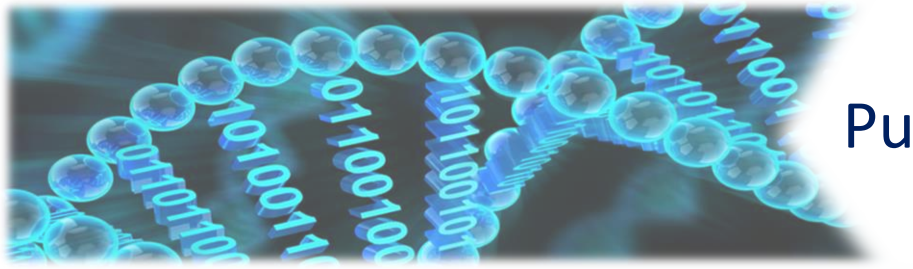
R skeleton – https://kendomaniac.github.io/docker4seq/articles/skeleton.html

Test Sets – https://kendomaniac.github.io/docker4seq/articles/docker4seq.html#test-sets

Workflow for mRNAseq analysis – https://kendomaniac.github.io/docker4seq/articles/docker4seq.html#rnaseq-workflow-howto

Workflow for miRNAseq analysis – https://kendomaniac.github.io/docker4seq/articles/docker4seq.html#mirnaseq-workflow

Workflow for ChIPseq analysis – https://kendomaniac.github.io/docker4seq/articles/docker4seq.html#chipseq-workflow



Publications

Kulkarni *et al. BMC Bioinformatics* 2018, **19**(Suppl 10):349
https://doi.org/10.1186/s12859-018-2296-x

BMC Bioinformatics

RESEARCH Open Access

Reproducible bioinformatics project: a community for reproducible bioinformatics analysis pipelines

Neha Kulkarni[1], Luca Alessandrì[1], Riccardo Panero[1], Maddalena Arigoni[1], Martina Olivero[2], Giulio Ferrero[3], Francesca Cordero[3*], Marco Beccuti[3†] and Raffaele A. Calogero[1*†]

From Italian Society of Bioinformatics (BITS): Annual Meeting 2017
Cagliari, Italy. 05-07 July 2017

Abstract

Background: Reproducibility of a research is a key element in the modern science and it is mandatory for any industrial application. It represents the ability of replicating an experiment independently by the location and the operator. Therefore, a study can be considered reproducible only if all used data are available and the exploited computational analysis workflow is clearly described. However, today for reproducing a complex bioinformatics analysis, the raw data and the list of tools used in the workflow could be not enough to guarantee the reproducibility of the results obtained. Indeed, different releases of the same tools and/or of the system libraries (exploited by such tools) might lead to sneaky reproducibility issues.

Results: To address this challenge, we established the *Reproducible Bioinformatics Project (RBP)*, which is a non-profit and open-source project, whose aim is to provide a schema and an infrastructure, based on docker images and R package, to provide reproducible results in Bioinformatics. One or more Docker images are then defined for a workflow (typically one for each task), while the workflow implementation is handled via R-functions embedded in a package available at github repository. Thus, a bioinformatician participating to the project has firstly to integrate her/his workflow modules into Docker image(s) exploiting an Ubuntu docker image developed ad hoc by RPB to make easier this task. Secondly, the workflow implementation must be realized in R according to an R-skeleton function made available by RPB to guarantee homogeneity and reusability among different RPB functions. Moreover she/he has to provide the R vignette explaining the package functionality together with an example dataset which can be used to improve the user confidence in the workflow utilization.

Conclusions: Reproducible Bioinformatics Project provides a general schema and an infrastructure to distribute robust and reproducible workflows. Thus, it guarantees to final users the ability to repeat consistently any analysis independently by the used UNIX-like architecture.

Keywords: Reproducible research, Docker, Whole transcriptome sequencing, microRNA sequencing, Chromatin Immuno precipitation sequencing, Community, Single nucleotide variants

* Correspondence: francesca.cordero@unito.it; raffaele.calogero@unito.it
[†]Marco Beccuti and Raffaele A. Calogero contributed equally to this work.
[3]Department of Computer Sciences, University of Torino, Torino, Italy
[1]Department of Molecular Biotechnology and Health Sciences, University of Torino, Torino, Italy
Full list of author information is available at the end of the article

Background

Recently Baker and Lithgow [1, 2] highlighted the problem of the reproducibility in research. Reproducibility criticality affects to different extent a large portion of the science fields [1]. Since nowadays bioinformatics plays an important role in many biological and medical studies [3], a great effort must be put to make such computational analyses reproducible [4, 5]. Reproducibility issues in bioinformatics might be due to the short half-life of the bioinformatics software, the complexity of the pipelines, the uncontrolled effects induced by changes in the system libraries, the incompleteness or imprecision in workflow description, etc. To deal with reproducibility issues in Bioinformatics Sandve [5] suggested ten good practice rules for the development and the utilization of a computational workflow (Table 1). A community that fulfills some of the rules suggested by Sandve is Bioconductor [6] project, which provides version control for a large amount of genomics/bioinformatics packages. In this way, old releases of any Bioconductor package are kept available for the users. However, Bioconductor does not cover all the steps of any possible bioinformatics workflow, e.g. in RNAseq wolkflow fastq trimming and alignment steps are generally done using tools not implemented in Bioconductor. BaseSpace [7, 8] and Galaxy [9] represent an example of both commercial and open-source cloud solutions, which partially fulfill Sandve's roles. Furthermore, the workflows implemented in such environments cannot be heavily customized, e.g. BaseSpace has strict rules for applications submission. Moreover, clouds applications have to cope with legal and ethical issues [10].

Galaxy instead implements the functional reproducibility level, i.e. the information about data and the utilized tools are saved in terms of meta-data, while RBP exploiting Docker framework provides also the computation reproducibility, i.e. the real image of the computation environment used to generate the date is stored.

Table 1 Good practice bioinformatics rules, derived from Sandve et al. [5]

1	For Every Result, Keep Track of How It Was Produced
2	Avoid Manual Data Manipulation Steps
3	Archive the Exact Versions of All External Programs Used
4	Version Control All Custom Scripts
5	Record All Intermediate Results, When Possible in Standardized Formats
6	For Analyses That Include Randomness, Note Underlying Random Seeds
7	Always Store Raw Data behind Plots
8	Generate Hierarchical Analysis Output, Allowing Layers of Increasing Detail to Be Inspected
9	Connect Textual Statements to Underlying Results
10	Provide Public Access to Scripts, Runs, and Results

Recently container technology, a lightweight Operation System (OS)-level virtualization, was explored in the area of Bioinformatics to make easier the distribution, the utilization and the maintenance of bioinformatics software [11–13]. Indeed, since applications and their dependencies are packaged together in the container image, the users have not to download and install all the dependencies required by an application, thus avoiding all the cases where the dependencies are not well documented or not available at all. Moreover, problems related to versions conflicts or updates of the system libraries do not occur, because the containers are isolated and frozen from the rest of the operating system.

Among the available container platforms, Docker (http://www.docker.com) is becoming de facto the standard environment to quickly compose, create, deploy, scale and oversee containerized applications under Linux. Its strengths are the high degree of portability, which allows users to register and share containers over various hosts in private and public repositories, and to achieve a more effective resource use and a faster deployment compared with other similar software.

In Menegidio [13], da Veiga [11] and Kim [12] the authors provide a large collection of bioinformatics tools containerized in a Docker image called BioContainers. However, a controlled and flexible framework to create and distribute bioinformatics reproducible workflow is not defined. Instead, projects like (https://snakemake.bitbucket.io) or Nextflow (https://www.nextflow.io) allow users to create reproducible and scalable data analyses specifying their own pipeline through a powerful metalanguage for workflow specification. However, the strong flexibility of these metalanguages can make difficult their utilization for users without advanced programming skills.

To cope with these aspects, we propose the implementation of the Reproducible Bioinformatics Project (RBP, http://reproducible-bioinformatics.org/), whose aims are (i) to distribute to the bioinformatics community docker-based applications under the reproducibility framework proposed by Sandve [5], and (ii) to provide to R bionformatics community an easier framework for the developing their own reproducible workflows.

The concept of BioContainers, described above, is different from RBP project. BioContainers provides pieces of software to be integrated in a workflow, as instead in RBP complete workflows are provided, e.g. gene/transcripts RNAseq, microRNA-sequencing (miRNA-seq), Chromatin Immuno Precipitation sequencing (ChIP-seq), DNA/RNAseq variant calling. RBP docker images not only include the specific software that give the name to the image, e.g. in bwa RBP docker image, bwa.2017.01, samtools, picard-tools, java and R, are also present.

RBP accepts simple docker implementations of *bioinformatics software* (e.g. a docker embedding bwa aligner tool), implementation of *complex pipelines* involving the use of multiple dockers images (e.g. a RNA-seq workflow providing all the steps for an analysis starting from the quality control of the fastq to differential expression), as well as *demonstrative workflows (*i.e. docker images embedding the full bioinformatics workflow used in a publication) intended to provide the ability to reproduce published data.

Methods

The Reproducible Bioinformatics Project (RBP) reference web page is http://reproducible-bioinformatics.org. The project is based on three modules (Fig. 1): (i) *docker4seq R package* (https://github.com/kendomaniac/docker4seq), (ii) *dockers images* (https://hub.docker.com/u/repbioinfo/), and (iii) *4SeqGUI* (https://github.com/mbeccuti/4SeqGUI).

Docker4seq package provides the interface between users and docker containers. *Docker4seq* is organized in two branches: stable and development. The transition between development and stable branch is done when a module (R function(s)/docker container(s)) fulfills the 10 rules suggested by Sandve [5] for the good bioinformatics practice (Table 1).

The function *skeleton.R* in docker4seq provides a prototype to build a docker controlling function. A tutorial on how to use the skeleton.R function is available in the section "How to be part of the Reproducible Bioinformatics project" at http://www.reproducible-bioinformatics.org/ and the skeleton.R is part of the devel branch of docker4seq (https://github.com/kendomaniac/docker4seq/tree/devel). The tutorial also embeds a description of the Ubuntu docker image called via skeleton.R. In the docker images repository *docker.io/repbioinfo* is available an Ubuntu image, which is the starting image used for the creation of all docker images developed by the RBP core team. Since, there are no specific software requirements for the docker images present in RBP, developers can use any linux image to build their own docker image.

Acknowledgments of the developer work is provided within the structure of the *skeleton.R.* In *skeleton.R* there is a field indicating developer affiliation and email for contacts.

Developer is free to decide to use this prototype or to adapt a different Linux docker distribution for his/her application. Docker images designed by the core developers of RBP are located in *docker.io/repbioinfo* (docker.com), the images developed by third parties can be instead placed in any public-access docker repository.

RBP requires that any operation, implying the use of any R/Bioconductor packages or the use of an external software, has to be implemented in a docker container. Only reformatting actions, e.g. table assembly, data reordering, etc., can be handled outside a docker image.

Any new RBP module (R function(s)/docker image(s)) must be associated with an explanatory vignette, accessible online as html document, and with a set of test data accessible online. Thus, all instruments needed to acquire confidence on module functionalities are provided to the final user.

Docker images are labelled with the extension YYYY.NN, where YYYY is the year of insertion in the stable version and NN a progressive number. YYYY changes only if any update on the program(s), implemented in the docker image, is done. This because any of such updates will affect the reproducibility of the workflow. Previous version(s) will be also available in the repository. NN refers to changes in the docker image, which do not affect the reproducibility of the workflow.

A new module can be submitted to the info@reproducible-bioinformatics.org and RBP core team will verify the compliance with Sandve [5] rules. Specifically, to guarantee the compliance with Sandve rules, RBP core team will check that:

- Each new workflow produces for each analysis step a log file, thus tracking how the results are produced (Sandve rule 1).
- All workflow/module steps are executed through scripts, thus avoiding manual data manipulation steps (Sandve rule 2).
- All computation events are executed within a docker container and the versions of the software

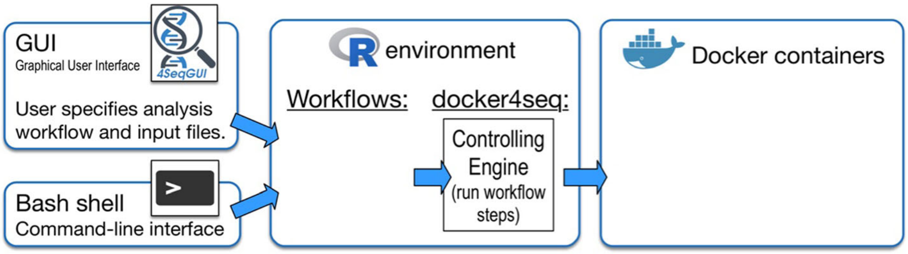


Fig. 1 Reproducible Bioinformatics Project structure

embedded in the docker image is shown as tag of the docker image (Sandve rule 3, 4).

- All intermediate results are available as part of the final results (Sandve rule 5).
- In case random seeds are used, they are recorded in a file and provided as part of final output of the module (Sandve rule 6).
- Raw data used to generate plots should be made available with plots (Sandve rule 7).
- Sandve rules 8 and 9 are not considered mandatory, because are mostly dependent from the workflow/module. The RBP core team will check if compliance to these rules will improve the overall quality of workflow/module output.
- License associated with the modules/workflows embedded in docker4seq must guarantee public access to the scripts and docker images (Sandve rule 10).

Rules 8 and 9, reported in Table 1, are not considered mandatory.

Ones validated, the R functions controlling the new module are inserted into *docker4seq* stable release. Partially validated modules will be placed in development branch and moved to stable one when compliance with Sandve's rules is fulfilled.

4SeqGUI is a Java based graphical interface to docker4seq functions. It is designed to provide a GUI to users having limited knowledge of R scripting. Currently the GUI embeds only general-purpose workflows, such as RNAseq, miRNA-seq and Chip-seq workflow.

Results

The stable branch of *docker4seq R package* contains all the R functions required to handle all the steps of RNA-seq workflow (Fig. 2a), ChIP-seq workflow (Fig. 2b), and miRNA-seq workflow (Fig. 2c). *Docker4seq* also provides a wrapper function for the *bcl2fastq* Illumina tool to convert the Illumina sequencer output in demultiplexed fastq files (Fig. 2). Then, the fastq files can be handled with any of the three different workflows. The counts table produced by RNAseq or miRNAseq workflows can be used to data visualization (*pca,* principal component analysis function), to evaluate the statistical power of the experiment (*experimentPower* function), to define the optimal sample size of the experiment for the detection of differentially expressed genes (*sampleSize* function) and to detect differentially expressed genes/transcripts (*wrapperDeseq2* function). Sample size/statistical power estimation of the experiment and differential expression are calculated respectively via RnaSeqSampleSize [14] and DESeq2 Bioconductor packages [15].

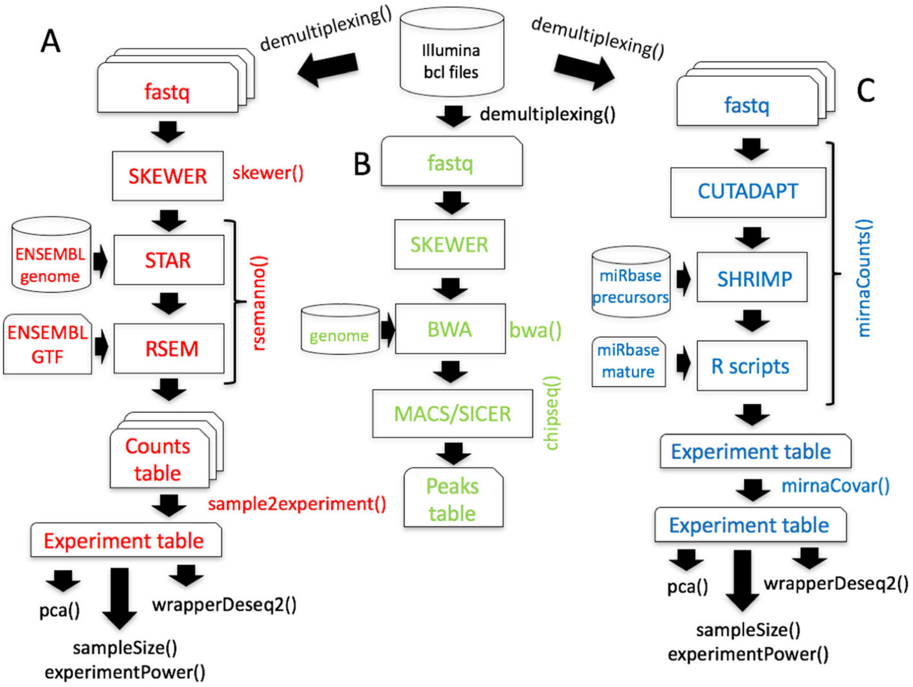


Fig. 2 Workflows available in the stable branch of docker4seq. **a** Whole transcriptome sequencing workflow, **b** ChIP sequencing workflow, and **c** miRNA sequencing workflow. The names followed by parenthesis are the docker4seq functions used to execute the analysis steps. Black indicate elements in common among more than one workflow

In the development branch, we work on three workflows (i) Patient Derived Xenograft (PDX) workflow, (ii) human small non-conding (snc) RNAs workflow, and (iii) B-cell clonality and Minimal Residual Disease detection.

In the first workflow we provide a pipeline for DNA (from EXOMEseq data) and RNA (from RNAseq data) somatic variant calling. The DNA variant calling workflow embeds the pre-processing procedure suggested by the GATK best practice (Fig. 3a). RNAseq data preparation for variant calling (Fig. 3c) requires the use of STAR 2 step procedure [16], which provides significantly increased sensitivity to novel splice junctions. Then, after sorting and duplicates marking, OPOSSUM [17] is used to remove intronic regions and to merge overlapping reads. We have also implemented a specific procedure (Fig. 3b), based on xenome software [18], to discriminate between human reads and mouse host reads in the sequences produced by the analysis of patients derived xenografts (PDX, [19]). As part of the somatic variant calling workflow we are implementing MUTECT 1 and 2 [20] (Fig. 4a) to call somatic variants as well as PLATYPUS [21] for extracting information of joined-samples Single Nucleotide Variants (SNVs)(Fig. 4b).

We are also expanding the RNAseq module adding the reference-free Salmon aligner [22], which employs less memory for the alignment task than STAR, but providing similar results [23].

The second workflow, used in the analysis described in the paper by Ferrero et al. [24], is focus on the analysis of sncRNAs as reported in Fig. 5. The quality of the FASTQ files are checked using FastQC software. The reads associated with good quality values are clipped from the adapter sequences using Cutadapt. The trimmed reads are then mapped against an in-house reference of human small RNA sequences composed of: (i) 1881 precursor miRNA sequences downloaded from miRBase (Release 21) (ii) 32,826 piRNA sequences from piRBase v1.0, and (iii) 5171 small RNA sequences from Database of Small Human non-coding RNAs (DASHR) database v 1.0 shorter than 80 bp.

The alignment is performed using the BWA algorithm. Small RNAs quantification is performed differently between

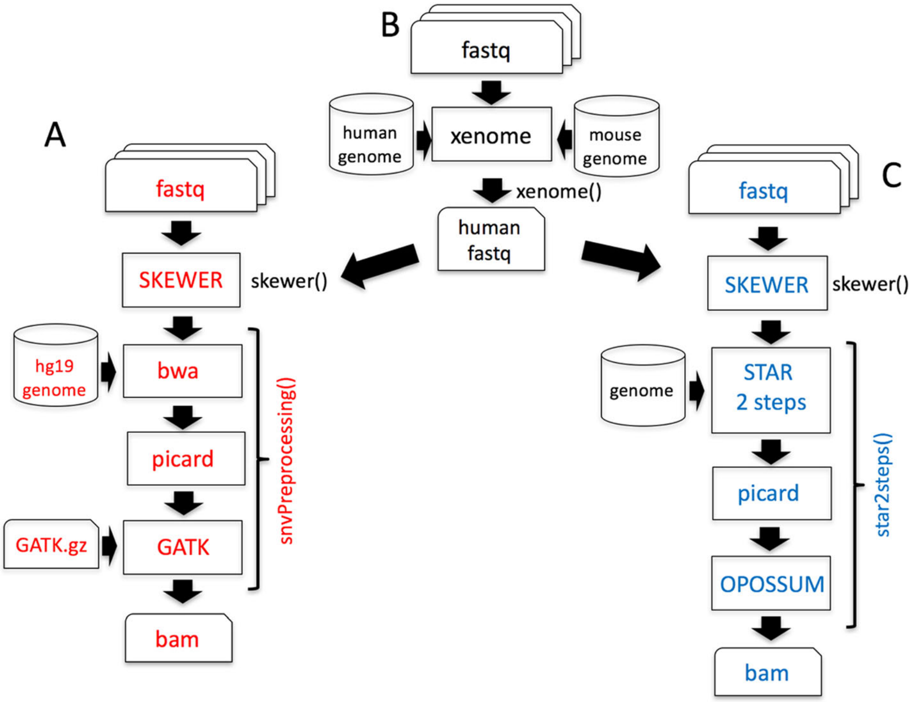


Fig. 3 Variant calling workflows under refinement in the development branch of docker4seq. **a** SNVs calling in DNA workflow. The function *snvPreprocessing* requires that users provides its own copy of the GATK software, because of Broad Institute license restrictions. This function returns a bam file sorted, with duplicates marked after GATK indel realignment and quality recalibration. **b** Data preprocessing for samples derived by Patient Derived Xenografths (PDX). The *xenome* function discriminates between the mouse host reads and the human tumor reads, then DNA or RNA SNV calling workflows can be applied. **c** SNVs calling in RNA workflow. The function *star2steps* generates a sorted bam, where duplicates are marked and processed by opossum for removal of intronic regions and merging of overlapping reads. The names followed by parenthesis are the docker4seq functions used to execute the analysis steps. Black indicate elements in common between more than one workflow

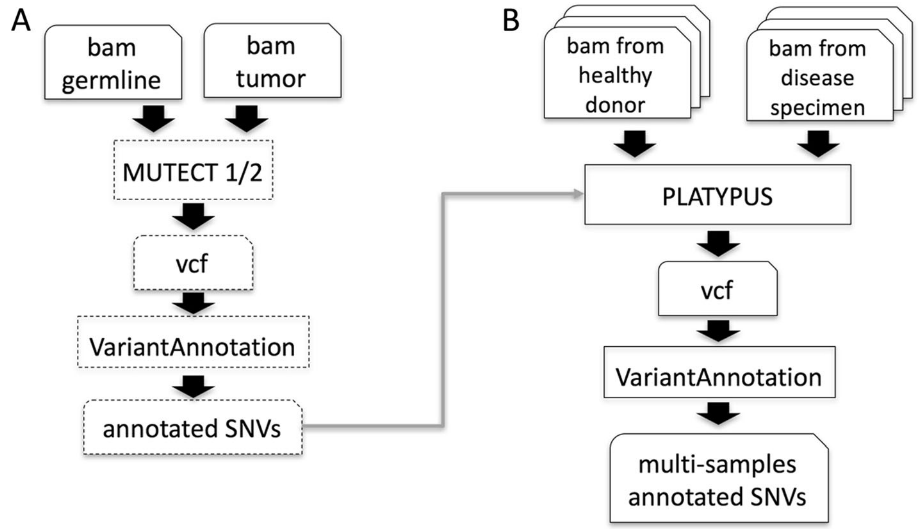


Fig. 4 Variant calling workflows under development in the development branch of docker4seq. **a** Somatic SNVs detection using GATK MUTECT 1 or 2. **b** Platypus based join mutations caller. Dashed blocks are not implemented, yet

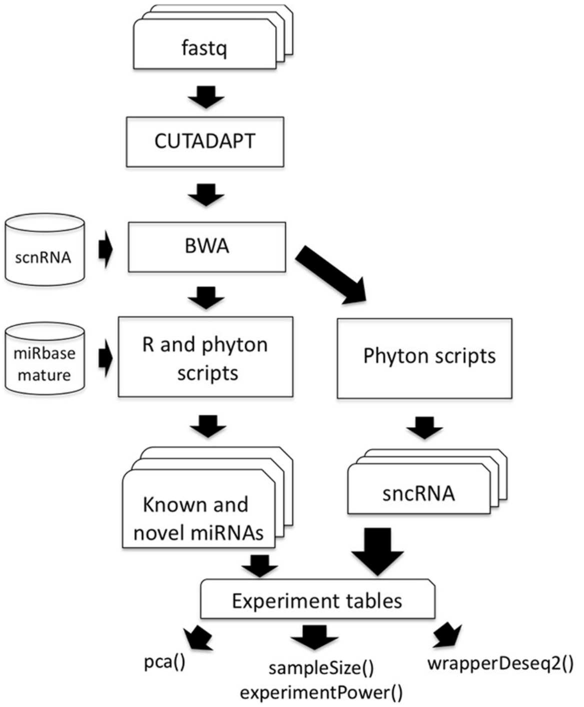


Fig. 5 sncRNA workflow. The sncRNA pipeline starts from a reference composed by the set of sncRNAs that contains all sncRNA characterized by a length minor than 80 bp. Then, two types of scripts are used one dedicated to the detection of known and novel microRNAs while the other is focused on sncRNAs

miRNAs and non miRNAs sncRNAs. The miRNA expression is quantify using two methods, called annotation-based or the position-based method respectively. In the annotat ion-based method, mature miRNAs expression quantification is performed by counting the read mapped on miRBase mature miRNA sequences using an GenomicRanges R package. Since not all miRNA mature sequences are annotated in miRBase, the position-based read count method is performed by considering the read mapping position within the precursor miRNA sequences. The result of the two quantification methods are merged into a final miRNA count matrix. In this matrix each mature miRNA not annotated in miRBase but quantified using the position-based method is reported with suffix *Novel*. Quantification of non miRNA annotations is performed counting the read alignment reported by BWA output sam files. The identification of Differentially Expressed sncRNAs is performed using Deseq2 package as reported in the RNAseq workflow.

The third workflow is based on the HashClone framework [25, 26] a new suite of bioinformatics tools providing B-cells clonality assessment and minimal residual disease (MRD) monitoring over time from deep sequencing data, was integrated in the *Docker4seq* package. In particular, a parallel version of the standard HashClone workflow (Fig. 6) was developed exploiting the docker architecture.

All the modules described above are implemented in 22 docker images deposited in the docker hub (https://hub.docker.com/u/repbioinfo/).

As part of the RBP we have also developed a GUI, 4Seq-GUI (https://github.com/mbeccuti/4SeqGUI). The GUI is implemented in JAVA and can be exploited to perform whole transcriptome sequencing workflow (Fig. 2a), ChIP

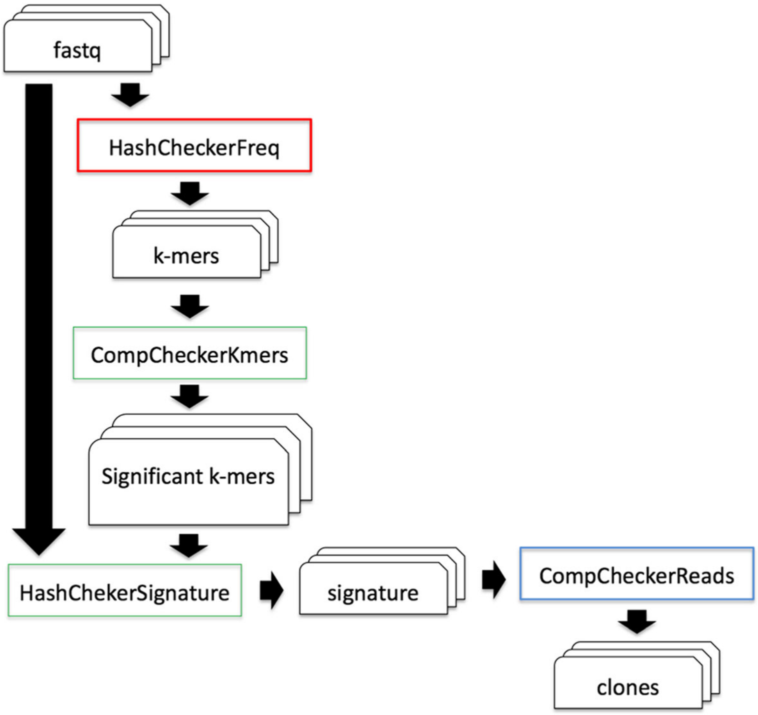


Fig. 6 HashClone pipeline. The HashClone strategy is organized in three steps: The first step (red box) is used to detect k-mer in all patients' samples. The second step (green box) focus on the generation of sequence signatures leading to the identification of the set of putative clones present in each of the patients' sample; the third step (blue box) is used to the characterization and evaluation of the cancer clones

sequencing workflow (Fig. 2b), and miRNA sequencing workflow (Fig. 2c).

Discussion

RBP core developers created frameworks for RNA/miRNA quantification and analysis. ChIPseq workflow was also developed and variant calling workflows for DNA and RNA are under active development. A peculiar feature of RBP is the acceptance of *demonstrative workflows*, i.e. bioinformatics procedures described in a biological/medical paper. A demonstrative workflow is wrapped in a docker image and it is supported by a tutorial, which describes step by step how the analysis is done to guarantee the reproducibility of published data.

Conclusions

Bioinformatics workflows are becoming an essential part of many research papers. However, absence of clear and well-defined rules on the code distribution make the results of most published researches unreproducible [27]. Recently, Almugbel and coworkers [28] described an interesting infrastructure to embed Bioconductor based packages. However, Bioconductor does not cover all steps of any possible bioinformatics workflow, thus providing a limited framework for developing complex pipelines. Differently, RBP represents a new instrument, which expands the idea of Almugbel [28], providing a more flexible infrastructure allowing the bioinformatics community to spread their work under the guidance of rules, which guarantee inter-laboratory reproducibility and do not limit docker implementations to Bioconductor packages. Moreover the RBP project, differently by others projects i.e. snakemake and nextflow, is specifically designed for the R community.

The RBP workflows are designed to work on a single machine with multi-cores, which do not need to be necessary a high-end server [29]. In [29] we describe that RNAseq, miRNA-seq and ChIP-Seq workflows (Fig. 2) can be executed efficiently on a consumer computer equipped with Intel i7 CPU (8 threads), 250 Gb SSD disk and 32 Gb of RAM. Recently, with the implementation of the reference free aligner Salmon [22] the minimal RAM requirements dropped to 8 Gb. This make possible the execution of the workflows available in RBP nearly any modern laptop with Linux operating system. Of course, a high-end server allow an higher level of parallelization in the analysis of multiple samples. The advantage of a high-end server become also evident in case of the analysis of large datasets, e.g. whole genome variant calling or thousands of RNAseq experiments.

A future work will be to extend our project to deal with cluster and cloud architectures. Two possible directions will be investigated (i) to exploit the swarm mode provided by docker considering each service as a "single-shoot" service, and (ii) to provide an automatic translation of our workflow specified in R into an equivalent workflow specified in snakemake format or in nextflow format.

Availability and requirements

Project name: Reproducible Bioinformatics Project.

Project home page: http://reproducible-bioinformatics.org

Operating system: UNIX-like.

Programming language: R.

Other requirements: docker version 17.05.0-ce or higher.

License: GPL.

Abbreviations

ChIP-seq: Chromatin immuno precipitation sequencing; miRNA-seq: microRNA sequencing; OS: Operation system; PCA: Principal component analysis; PDX: Patient derived xenograft; RBP: Reproducible bioinformatics project; sncRNA: Human small non-conding RNA; SNVs: Single nucleotide variants

Funding

EPIGEN FLAG PROJECT (responsible: RAC) supported this work including the publication charge. This work is also supported by CRT project RF = 2017.2025 (principal investigator: FC).

About this supplement

This article has been published as part of *BMC Bioinformatics* Volume 19 Supplement 10, 2018: Italian Society of Bioinformatics (BITS): Annual Meeting 2017. The full contents of the supplement are available online at https://bmcbioinformatics.biomedcentral.com/articles/supplements/volume-19-supplement-10.

Authors' contributions

NK and LA equally contributed to the development of miRNA workflow and all the other tools. RP redefined the ChIPseq workflow. FC and GF developed the miRNAseq, scnRNA, and ParallelHashClone workflows. MA and MO performed applications testing. FC, MB and RAC developed the rules to submit tools and workflows to the Reproducible Bioinformatics community. RAC and MB equally supervised the overall work. All the authors have read and approved the final manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

[1]Department of Molecular Biotechnology and Health Sciences, University of Torino, Torino, Italy. [2]Department of Oncology, University of Torino, Candiolo, Italy. [3]Department of Computer Sciences, University of Torino, Torino, Italy.

Published: 15 October 2018

References

1. Baker M. 1,500 scientists lift the lid on reproducibility. Nature. 2016; 533(7604):452–4.
2. Lithgow GJ, Driscoll M, Phillips P. A long journey to reproducible results. Nature. 2017;548(7668):387–8.
3. Searls DB. The roots of bioinformatics. PLoS Comput Biol. 2010;6(6): e1000809.
4. Kanwal S, Khan FZ, Lonie A, Sinnott RO. Investigating reproducibility and tracking provenance - a genomic workflow case study. BMC Bioinf. 2017; 18(1):337.
5. Sandve GK, Nekrutenko A, Taylor J, Hovig E. Ten simple rules for reproducible computational research. PLoS Comput Biol. 2013;9(10): e1003285.
6. Gentleman RC, Carey VJ, Bates DM, Bolstad B, Dettling M, Dudoit S, Ellis B, Gautier L, Ge Y, Gentry J, et al. Bioconductor: open software development for computational biology and bioinformatics. Genome Biol. 2004;5(10):R80.
7. Colombo AR, Triche JT Jr, Ramsingh G. Arkas: Rapid reproducible RNAseq analysis. F1000Res. 2017;6:586.
8. Van Neste C, Gansemans Y, De Coninck D, Van Hoofstat D, Van Criekinge W, Deforce D, Van Nieuwerburgh F. Forensic massively parallel sequencing data analysis tool: implementation of MyFLq as a standalone web- and Illumina BaseSpace((R))-application. Forensic Sci Int Genet. 2015;15:2–7.
9. Digan W, Countouris H, Barritault M, Baudoin D, Laurent-Puig P, Blons H, Burgun A, Rance B. An architecture for genomics analysis in a clinical setting using galaxy and Docker. Gigascience. 2017;6(11):1-9.
10. Dove ES, Joly Y, Tasse AM, Public Population Project in G, Society International Steering C, International Cancer Genome Consortium E, Policy C, Knoppers BM. Genomic cloud computing: legal and ethical points to consider. Eur J Hum Genet : EJHG. 2015;23(10):1271–8.
11. da Veiga LF, Gruning BA, Alves Aflitos S, Rost HL, Uszkoreit J, Barsnes H, Vaudel M, Moreno P, Gatto L, Weber J, et al. BioContainers: an open-source and community-driven framework for software standardization. Bioinformatics. 2017;33(16):2580–2.
12. Kim B, Ali T, Lijeron C, Afgan E, Krampis K. Bio-Docklets: virtualization containers for single-step execution of NGS pipelines. Gigascience. 2017;6(8):1–7.
13. Menegidio FB, Jabes DL, Costa de Oliveira R, Nunes LR. Dugong: a Docker image, based on Ubuntu Linux, focused on reproducibility and replicability for bioinformatics analyses. Bioinformatics. 2017;34(3):514-5.
14. Ching T, Huang S, Garmire LX. Power analysis and sample size estimation for RNA-Seq differential expression. RNA. 2014;20(11):1684–96.
15. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. 2014;15(12):550.
16. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, Batut P, Chaisson M, Gingeras TR. STAR: ultrafast universal RNA-seq aligner. Bioinformatics. 2013;29(1):15–21.
17. Oikkonen L, Lise S. Making the most of RNA-seq: pre-processing sequencing data with opossum for reliable SNP variant detection. Wellcome Open Res. 2017;2:6.
18. Conway T, Wazny J, Bromage A, Tymms M, Sooraj D, Williams ED, Beresford-Smith B. Xenome--a tool for classifying reads from xenograft samples. Bioinformatics. 2012;28(12):i172–8.
19. Siolas D, Hannon GJ. Patient-derived tumor xenografts: transforming clinical samples into mouse models. Cancer Res. 2013;73(17):5315–9.
20. Cibulskis K, Lawrence MS, Carter SL, Sivachenko A, Jaffe D, Sougnez C, Gabriel S, Meyerson M, Lander ES, Getz G. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. Nat Biotechnol. 2013;31(3):213–9.
21. Rimmer A, Phan H, Mathieson I, Iqbal Z, Twigg SRF, Consortium WGS, Wilkie AOM, McVean G, Lunter G. Integrating mapping-, assembly- and haplotype-based approaches for calling variants in clinical sequencing applications. Nat Genet. 2014;46(8):912–8.
22. Patro R, Duggal G, Love MI, Irizarry RA, Kingsford C. Salmon provides fast and bias-aware quantification of transcript expression. Nat Methods. 2017; 14(4):417–9.
23. Zhang C, Zhang B, Lin LL, Zhao S. Evaluation and comparison of computational tools for RNA-seq isoform quantification. BMC Genomics. 2017;18(1):583.

24. Ferrero G, Cordero F, Tarallo S, Arigoni M, Riccardo F, Gallo G, Ronco G, Allasia M, Kulkarni N, Matullo G, Vineis P, Calogero RA, Pardini B, Naccarati A. Small non-coding RNA profiling in human biofluids and surrogate tissues from healthy individuals: description of the diverse and most represented species. Oncotarget. 2018;9:3097–111.
25. Beccuti M, Genuardi E, Romano G, Monitillo L, Barbero D, Boccadoro M, Ladetto M, Calogero R, Ferrero S, Cordero F. HashClone: a new tool to quantify the minimal residual disease in B-cell lymphoma from deep sequencing data. BMC Bioinformatics. 2017;18(1):516.
26. Romano G, Genuardi R, Calogero R, Ferrero S. ParallelHashClone: a parallel implementation of HashClone suite dor clonality assessment from NGS data. In: P26th Euromicro International Conference on Parallel, Distribuited and Netwrok-based Processing(PDP) 2018, Cambridge, UK, March 21-23, 2018.
27. Hothorn T, Leisch F. Case studies in reproducibility. Brief Bioinform. 2011; 12(3):288–300.
28. Almugbel R, Hung LH, Hu J, Almutairy A, Ortogero N, Tamta Y, Yeung KY. Reproducible Bioconductor workflows using browser-based interactive notebooks and containers. J Am Med Inform Assoc. 2017;25(1):4-12.
29. Beccuti M, Cordero F, Arigoni M, Panero R, Amparore EG, Donatelli S, Calogero RA. SeqBox: RNAseq/ChIPseq reproducible analysis on a consumer game computer. Bioinformatics. 2017;34(5):871-2.

Research Paper

Small non-coding RNA profiling in human biofluids and surrogate tissues from healthy individuals: description of the diverse and most represented species

Giulio Ferrero[1,2,*], Francesca Cordero[1,3,*], Sonia Tarallo[3], Maddalena Arigoni[4], Federica Riccardo[4], Gaetano Gallo[5,6], Guglielmo Ronco[7], Marco Allasia[8], Neha Kulkarni[4], Giuseppe Matullo[3,9], Paolo Vineis[3,10], Raffaele A. Calogero[4], Barbara Pardini[3,9,*] and Alessio Naccarati[3,11,*]

[1]Department of Computer Science, University of Turin, Turin, Italy

[2]Department of Clinical and Biological Sciences, University of Turin, Turin, Italy

[3]Italian Institute for Genomic Medicine, IIGM (formerly Human Genetics Foundation, HuGeF), Turin, Italy

[4]Molecular Biotechnology Center, Department of Biotechnology and Health Sciences, University of Turin, Turin, Italy

[5]Department of Medical and Surgical Sciences, University of Catanzaro, Catanzaro, Italy

[6]Department of Colorectal Surgery, Clinica S. Rita, Vercelli, Italy

[7]Center for Cancer Epidemiology and Prevention, AO City of Health and Science, Turin, Italy

[8]Department of Surgical Sciences, University of Turin and Città della Salute e della Scienza, Turin, Italy

[9]Department of Medical Sciences, University of Turin, Turin, Italy

[10]MRC-HPA Centre for Environment and Health, School of Public Health, Imperial College London, London, United Kingdom

[11]Department of Molecular Biology of Cancer, Institute of Experimental Medicine, Prague, Czech Republic

[*]These authors contributed equally to this work

Correspondence to: *Alessio Naccarati,* ***email:*** *alessio.naccarati@iigm.it*

Keywords: *next-generation sequencing; small non-coding RNA profiling; microRNAs; non-invasive biomarkers; surrogate tissues*

Received: *September 06, 2017* ***Accepted:*** *November 15, 2017* ***Published:*** *December 14, 2017*

ABSTRACT

The role of non-coding RNAs in different biological processes and diseases is continuously expanding. Next-generation sequencing together with the parallel improvement of bioinformatics analyses allows the accurate detection and quantification of an increasing number of RNA species. With the aim of exploring new potential biomarkers for disease classification, a clear overview of the expression levels of common/unique small RNA species among different biospecimens is necessary. However, except for miRNAs in plasma, there are no substantial indications about the pattern of expression of various small RNAs in multiple specimens among healthy humans.

By analysing small RNA-sequencing data from 243 samples, we have identified and compared the most abundantly and uniformly expressed miRNAs and non-miRNA species of comparable size with the library preparation in four different specimens (plasma exosomes, stool, urine, and cervical scrapes).

Eleven miRNAs were commonly detected among all different specimens while 231 miRNAs were globally unique across them. Classification analysis using these miRNAs provided an accuracy of 99.6% to recognize the sample types. piRNAs and tRNAs were the most represented non-miRNA small RNAs detected in all specimen types that were analysed, particularly in urine samples. With the present data, the

most uniformly expressed small RNAs in each sample type were also identified. A signature of small RNAs for each specimen could represent a reference gene set in validation studies by RT-qPCR.

Overall, the data reported hereby provide an insight of the constitution of the human miRNome and of other small non-coding RNAs in various specimens of healthy individuals.

INTRODUCTION

The discovery of many stable extracellular small RNAs has changed our view of gene expression regulation, including the role that these molecules may play in several complex processes previously partially understood such as cell-to-cell communication [1]. In this respect, with an astonishing number of publications in the last decade, microRNAs (miRNAs) represent the most explored small non-coding RNA (sncRNA) species in humans [2]. A large number of studies has demonstrated that cellular and extracellular miRNA altered expression is associated with a wide variety of diseases, including cancer [3, 4]. However, little is known about the presence within the same matrix of other common species of sncRNAs such as piwi-interacting RNAs (piRNAs), small nucleolar RNAs (snoRNAs), tRNAs etc. All these versatile RNA species are known to be key components of molecular interactions and gene regulation in eukaryotes [5].

The field of circulating extracellular RNA molecules is rapidly growing thanks to the implementation of Next-Generation Sequencing (NGS) technologies and bioinformatics solutions that analyze the huge amount of data released from sequencing. With such high-throughput approach, all extracellular RNAs can be quantified and tested as potential sources of new diagnostic and therapeutic biomarkers in many different types of biological samples [6]. To achieve this, RNA-Sequencing (RNA-Seq) has emerged as a powerful tool in transcriptomics, gene expression profiling and biomarker discovery. Sequencing cell-free nucleic acids from liquid biopsies additionally provides exciting possibilities for molecular diagnostics, and might help establish disease-specific biomarker signatures [7]. Lower complexity, not known post-processing modifications, simple detection and amplification methods, tissue-restricted expression profiles, and sequence conservation between humans and model organisms make extracellular miRNAs and other sncRNAs ideal candidates for non-invasive biomarkers to reflect and study various physiopathological conditions in the body [8]. It is possible to extract and quantify high-quality sncRNAs from a wide range of cell and tissue sources, including cell lines, fresh and formalin-fixed paraffin-embedded tissues, plasma, serum, urine and other body fluids [8–10]. Despite this increasing interest, the field is still largely in an exploratory and descriptive phase. There are no standardized methods for sample collection, isolation, or analysis. There is also no general agreement on the terms for a good quality sample definition, and each specimen (body fluid or surrogate tissue) under various disease/injury conditions are likely to have diverse contents and different criteria for quality assessment [7, 11]. A growing number of isolation methods for profiling circulating extracellular RNA molecules have been developed but still, there is no gold standard for the most efficient inclusive or selective protocols [6]. However, the complexity of the small RNA-Seq workflow bears challenges and biases that researchers need to be aware of, in order to generate high-quality data [12].

The creation of large repositories including data from different human specimens, isolation methods, detection platforms, and analysis tools is essential to increase our understanding of the extent and types of extracellular RNA material present in different body fluids/surrogate tissues. At present, there are few large datasets describing the extracellular contents in biofluid samples from healthy controls [13–17]. Besides, previous studies on extracellular sncRNAs have investigated very small numbers or pooled samples with the purpose of identifying a specific class of RNAs [18]. The largest investigations of samples focused almost exclusively on miRNAs, with the main limitation of measuring either only targeted miRNAs in large numbers of individuals or the whole known miRNome in very small populations. In a recent work it has been described the largest group of plasma-based miRNAs and the first broadest variety of extracellular (non-miRNA) sncRNAs in a large population [15]. In another similar work, authors profiled the small RNA (16–32 nts) payload of human biofluids by NGS. Extracellular RNAs were isolated from plasma, urine and saliva samples from 55 young male athletes and sequenced to establish a sncRNA pattern at steady state [6].

In the present study, we investigated pattern and expression levels of miRNAs and other sncRNAs of comparable size in four different biospecimens representing ideal surrogate tissues for diagnostic and screening programs. Specifically, we analysed data from small RNA-Seq from 125 plasma-derived exosomes, 48 urine, 31 cervical scrapes, and 39 stool samples collected from healthy subjects. For cervical scrapes and stool, this is the first study investigating sncRNAs by NGS. In addition, urine and stool samples were paired with those from plasma collected from the same subjects.

RESULTS

Overview of study samples and pipeline analysis

We analysed small RNA-Seq data of RNA extracted from exosomes from 125 plasma samples of healthy donors derived from three different studies (respectively 39 for the Study 1, 46 for the Study 2, and 40 for the Study 3) (Materials and Methods). Additionally, sequencing was performed on RNA from 39 faecal samples (Study 1), 48 urine samples (Study 2), and cervical scrapes from 31 Human Papilloma Virus (HPV) negative women. Some of the plasma sample donors provided at the same occasion a sample of stool (39 from Study 1) or urine (46 from Study 2).

Total RNA was isolated from samples with specific kits for each type of specimens while library preparation for small RNA-Seq was performed adopting the same kit and protocol. Libraries were run at the same sequencing facility. Finally, all bioinformatics analyses (i.e. pre-processing of raw data) were performed following the same pipeline by the same operator.

To explore the landscape of sncRNA expression levels in different biospecimens, we designed a computational strategy for small RNA-Seq data analysis (Figure 1A). We updated the miRNA analysis pipeline published by our group [19] by adding a second phase focused on the analysis of small RNA-Seq reads unmapped against the human miRNome (Materials and Methods).

Initially, small RNA-Seq datasets were pre-processed and quality controlled to remove adapter sequences and low-quality reads. The processing information about the 243 datasets analysed is provided in Supplementary Table 1A and 1B. Quality check confirmed that were no reads shorter than 15 nucleotides and the rate of low quality reads (quality score < 30) was on average below 8%, with urine and stool samples providing the best rates (<1%).

Identification of miRNAs and non-miRNA sncRNAs

miRNA mapping analysis showed remarkable differences among specimens for read alignment rates (Figure 1B, 1C and Supplementary Figure 1A). Consistently with the highest rates of read alignment (Figure 1C), urine samples were generally associated with a high number of reads (median reads = 12.38 million) followed by plasma exosomes (median reads = 11.34 million), stool (median reads = 4.88 million), and cervical scrapes (median reads = 4.13 million) (Supplementary Figure 2A).

Datasets from plasma exosome and urine samples were characterized by the highest miRNA alignment rates (16.3% and 11.0%, of reads aligned, respectively) while datasets from stool and cervical scrape samples were associated on average with low miRNome alignment rates (0.7% and 1.2% of reads aligned, respectively).

Of the 1,823 miRNA annotations from miRBase, a range from 19.9% (cervical scrapes) to 73.8% (plasma exosomes *study* 2) of human miRNAs were detected in all the investigated specimens. A median of 58.61% of miRBase annotations were detected across the four specimen types. Specifically, miR-486-5p was the most expressed miRNA in plasma exosomes samples (median reads = 180,173 reads) while miR-320a (median reads = 198 reads), miR-6813-5p (median reads = 5,911 reads), and miR-30a-5p (median reads = 25,910 reads) were the highest expressed in cervical scrapes, stool, and urine, respectively (Supplementary Figure 2B).

Since a large fraction of sequencing reads did not map on miRNome (Supplementary Table 1A), the alignment analysis was extended to other candidate sncRNA annotations by initially remapping reads on the human genome. Then, mapped reads were assigned to sncRNA annotations quantifiable using our size selection criterion. These annotations included sncRNAs annotated in GENCODE v24 database [20] (transcript length ≤70 bp) as well as piRNA (average length 31±1 bp) and tRNA (average length 74±7 bp) species annotated in the Database of Small Human non-coding RNAs (DASHR) release 1 [21] (Supplementary Table 1C). The alignment rates observed were higher for cervical scrapes (88.4%) followed by urine (81.1%), and plasma exosome samples (69.5%). As expected, stool datasets were associated with the lowest alignment rate on the human genome (28.1%) consistently with the presence of microbiome RNAs and other RNAs introduced by the diet, contributing to the large fraction of faecal RNA content (Supplementary Table 1A and Supplementary Figure 1A). In urine, most reads were assigned to piRNA (44.5%) or tRNA annotations (45.1%). Conversely, in the other specimens, a low assignment rate was observed ranging 1.8–3.4% for piRNAs and 1.0–3.3% for tRNAs, respectively (Supplementary Figure 1B). Homologous piRNAs annotated to different loci were associated with the same number of reads across samples.

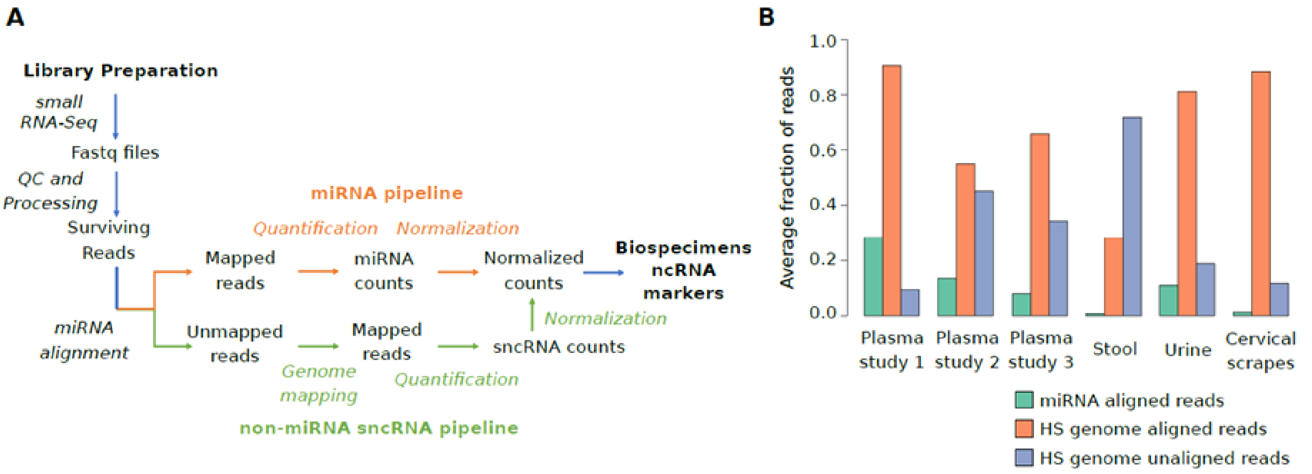
Common and specific miRNAs among different specimens

Considering the individual datasets from plasma exosome samples, it was evident a study-specific influence on the read alignment distribution with samples from the *study 1* characterized, on average, by the overall highest alignment on miRNome annotations (28.3% aligned reads). However, PCA on miRNAs and other sncRNA annotations expressed in at least one study (within study median number of reads >20) showed a distinct cluster formed by all plasma exosome samples with respect to other biospecimens (Supplementary Figure 1C). A comparable result was obtained by computing a pairwise correlation analysis: datasets from the three plasma exosome studies clustered together and were clearly separated from the others (Supplementary Figure 1D). Given the results from the PCA and correlation analyses, plasma exosome samples from the three studies were merged into a single group after read count correction with Surrogate Variable Analysis (SVA). The identification of pattern of miRNAs detectable in the different specimens was performed by considering miRNAs characterized by a median of normalized reads higher than 20 in at least one specimen. Using this threshold, cumulatively, 394 miRNAs were quantified in at least one specimen (Figure 2A, Supplementary Table 2A). Eleven miRNAs were identified as commonly detectable in all types of specimens: miR-320a, miR-589-5p, miR-636, miR-1273a, miR-3960, miR-4419a, miR-4497, miR-4709-5p, miR-4792, miR-7641-1, and miR-7641-2.

Functional enrichment analysis of validated target genes of the 11 shared miRNAs revealed biological processes related to mRNA translation and transcription including translational initiation (GO:0006413, $p = 1.9 \times 10^{-8}$) or positive regulation of transcription, DNA-templated (GO:0045893, $p = 4.2 \times 10^{-7}$) (Supplementary Table 2B).

Plasma exosome samples were characterized by the highest number of specimens-specific miRNAs (155 miRNAs) followed by stool (55 miRNAs), urine (22 miRNAs), and cervical scrape samples (one miRNA) (Figure 2A, 2B). Considering only the specimen-specific miRNAs, miR-122-5p was the most expressed in plasma exosome samples (median reads = 32,512 reads) while miR-655-5p (median reads = 792 reads), miR-204-5p (median reads = 750 reads) and miR-4741 (median reads = 28 reads) were the most abundantly expressed in stool, urine, and cervical scrapes, respectively (Supplementary Figure 2C).

PCA analysis of the highly-expressed sets of miRNAs showed a good accuracy in the classification of different biospecimens (Figure 2C). To identify the discriminative miRNAs in the specimen classification, we also performed a classification and attribute selection



C

Reads category (millions)	Plasma study 1 - Avg (Min – Max)	Plasma study 2 - Avg (Min – Max)	Plasma study 3 - Avg (Min – Max)
Raw reads	6.86 (0.47 - 15.01)	14.57 (0.01 - 43.87)	10.20 (0.57 - 25.49)
Surviving reads after cutadapt	5.87 (0.44 - 13.63)	10.72 (0.01 - 22.50)	5.00 (0.11 - 12.91)
miRnome mapped reads	1.58 (0.10 - 4.80)	1.75 (0.00 - 8.27)	0.36 (0.01 - 1.23)
miRNome unmapped reads mapped on hg38 genome	3.90 (0.31 - 9.19)	4.85 (0.01 - 13.45)	3.03 (0.06 - 7.66)
Gencode v24 assigned reads	3.23 (0.28 - 8.49)	0.87 (0.00 - 2.85)	0.43 (0.02 - 1.10)
DASHR piRNA assigned reads	0.06 (0.01 - 0.16)	0.08 (0.00 - 0.21)	0.06 (0.00 - 0.65)
DASHR tRNA assigned reads	0.04 (0.00 - 0.13)	0.03 (0.00 - 0.18)	0.04 (0.00 - 0.70)

Reads category (millions)	Stool - Avg (Min – Max)	Urine - Avg (Min – Max)	Cervical scrapes - Avg (Min – Max)
Raw reads	14.66 (0.22 - 70.41)	10.65 (1.21 - 48.88)	2.06 (0.21 - 14.96)
Surviving reads after cutadapt	11.33 (0.17 - 61.33)	7.09 (0.34 - 35.00)	0.76 (0.04 - 5.63)
miRnome mapped reads	0.07 (0.00 - 0.44)	0.42 (0.03 - 1.77)	0.01 (0.00 - 0.09)
miRNome unmapped reads mapped on hg38 genome	3.21 (0.05 - 15.53)	5.44 (0.15 - 26.99)	0.67 (0.04 - 5.36)
Gencode v24 assigned reads	0.39 (0.01 - 1.69)	0.77 (0.03 - 3.55)	0.11 (0.01 - 0.80)
DASHR piRNA assigned reads	0.05 (0.00 - 0.47)	3.25 (0.02 - 18.79)	0.03 (0.00 - 0.30)
DASHR tRNA assigned reads	0.05 (0.00 - 0.44)	3.20 (0.02 - 18.46)	0.03 (0.00 - 0.34)

Figure 1: (**A**) Schematic representation of the computational pipeline applied in the analysis of small RNA-Seq dataset from healthy individuals. The modules of the pipeline designed for miRNAs and other sncRNAs are depicted in orange and green, respectively. (**B**) Bar plot showing for each specimen, the average number of sequencing reads aligned to miRNA annotations (green), unmapped on miRNA annotations but mapped on human genome (red), and unmapped on both miRNA annotations and the human genome (blue). (**C**) Table reporting the average, minimum, and maximum number of reads (in million) composing the starting datasets, aligned in the different analysis phases, or assigned to specific RNA annotations. HS= Homo sapiens.

analysis. Using a Random Forest classifier, we obtained an accuracy of 99.6% with only one sample incorrectly classified (Supplementary Table 2C). All the miRNAs analysed were associated with a high chi-square statistic (merit) in the attribute selection analysis with miR-204-5p, miR-5698, and miR-335-3p associated with the highest merit (Supplementary Table 2D).

For a subset of patients, paired data from plasma and stool samples or from plasma and urine samples were available allowing a comparison between expression levels of sncRNAs in the different specimens from the same subject. As reported in Supplementary Table 2E, 2F, a low co-expression was generally observed either between plasma-stool or plasma-urine samples. The only exception was miR-3665 which was characterized by a positive correlation between plasma and urine samples ($r = 0.59$, $p = 2.0 \times 10^{-5}$).

Prediction of candidate miRNA isoforms (isomiRs) was also performed using our datasets. As reported in Supplementary Table 2G, 832 isomiRs associated with more than 20 supporting reads in at least one specimen type were detected. Overall, 94.4% of isomiRs were detected in plasma exosome or urine samples consistently with the higher number of aligned reads in these samples. The isomiRs with the highest number of supporting reads were a 3′ variant of miR-486/miR-486-2 in plasma samples, a 5′ variant of miR-934 in urine sample, a 5′ variant of miR-7704 in cervical scrapes, and a 3′ variant of miR-583 in stool samples. Among the previously identified 11 common miRNAs, eight were associated to an isomiR predicted in only one or two types of specimens (particularly in plasma or urine samples) (Supplementary Table 2H).

Expression pattern of other sncRNAs

Cumulatively, 615 non-miRNA sncRNAs were quantified in at least one specimen. Of this set of annotations, 112 sncRNAs were commonly detected in all the analysed sample types (Figure 3A and Supplementary Table 3A). Coherently with the highest alignment rates, piRNAs were the most represented type of sncRNAs in urine, plasma exosomes, and stool (Supplementary Figure 2D). Urine samples emerged as the specimen characterized by the highest piRNA and tRNA contents (Supplementary Figure 1B). Among the other sncRNAs identified there were tRNAs, mitochondrial RNAs, and snoRNAs particularly in plasma exosomes. Consistently,

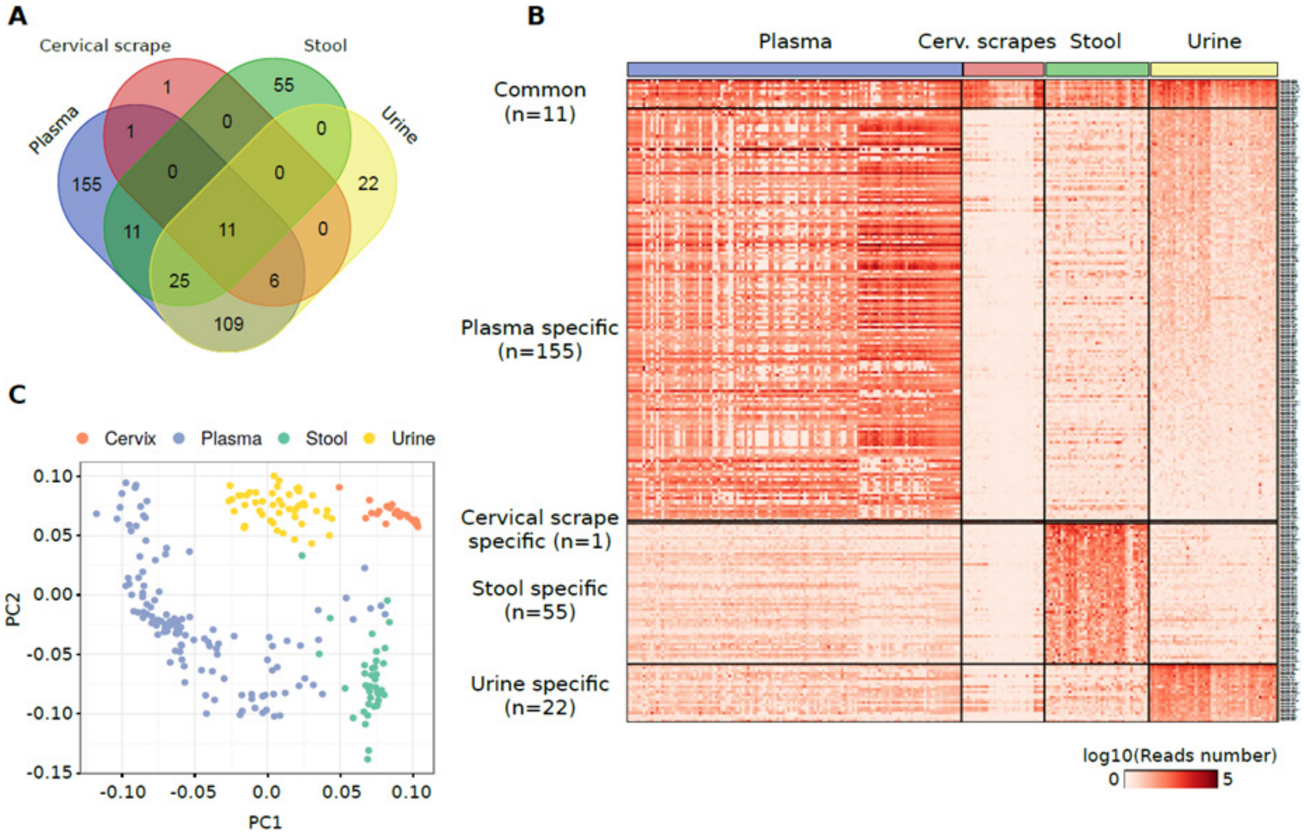


Figure 2: (**A**) Venn diagram reporting the number of miRNAs detected in different specimens from healthy individuals and their overlap. (**B**) Heat map showing the log10 number of normalized reads supporting the miRNAs specifically detected in one specimen or commonly detected among them. (**C**) PCA plot showing the small RNA-Seq datasets separation obtained using miRNAs detected in samples analysed.

considering those sncRNAs specific of each specimen, the highest number of sncRNAs was identified in urine (*n* = 127) and the same were grouped substantially apart from the other datasets in a PCA analysis using the sncRNA expression levels (Figure 3B).

PiR-31068 was the most abundant molecule in urine samples (Supplementary Figure 2D). The tRNA chr1.tRNA2-GlyCCC showed the highest expression levels among the sncRNAs specific in urine samples (median reads = 419 reads) while piR-43137 was the most abundant plasma exosome-specific sncRNA (median reads = 366 reads), and piR-36705 the most abundant stool-specific sncRNA (median reads = 131 reads) (Figure 3C and Supplementary Figure 2F). No specific sncRNAs of cervical scrapes were identified.

The specificity of these sets of sncRNAs was confirmed using a Random Forest classification algorithm which exactly classified 236 samples out of 243 (97.1%) (Supplementary Table 3B). The attribute selection analysis evidenced tRNAs chr19.tRNA2-GlyTCC, chr2.tRNA12-PseudoCTC, and chr6.tRNA150-MetCAT as the sncRNAs with the highest *merit* in the classification (Supplementary Table 3C).

Regression analysis between paired plasma exosome and stool samples or plasma exosome and urine samples from the same individuals showed a low coherent expression for sncRNAs detected (Supplementary Table 3D, 3E).

Assessing inter-individual variability in sncRNA expression in each specimen type

Independently of the extensive intrinsic variability among subject's extracellular RNA levels for each specimen, we selected the highly abundant sncRNAs with the lowest variable expression levels (i.e. potential reference sncRNAs) across all subjects. To achieve this, the highly-expressed miRNAs and sncRNAs specifically detected in plasma exosomes, stool, or urine (Figure 2A and 3A) characterized by the smallest expression variation in each specimen were identified by computing the median and the Median Absolute Deviation (MAD) of the expression levels (Supplementary Tables 2I and 3F). Specifically, the analysis highlighted miR-142-5p, miR-655-5p, and miR-196a-1-5p as potential *reference* miRNAs in plasma exosomes, stool, and urine,

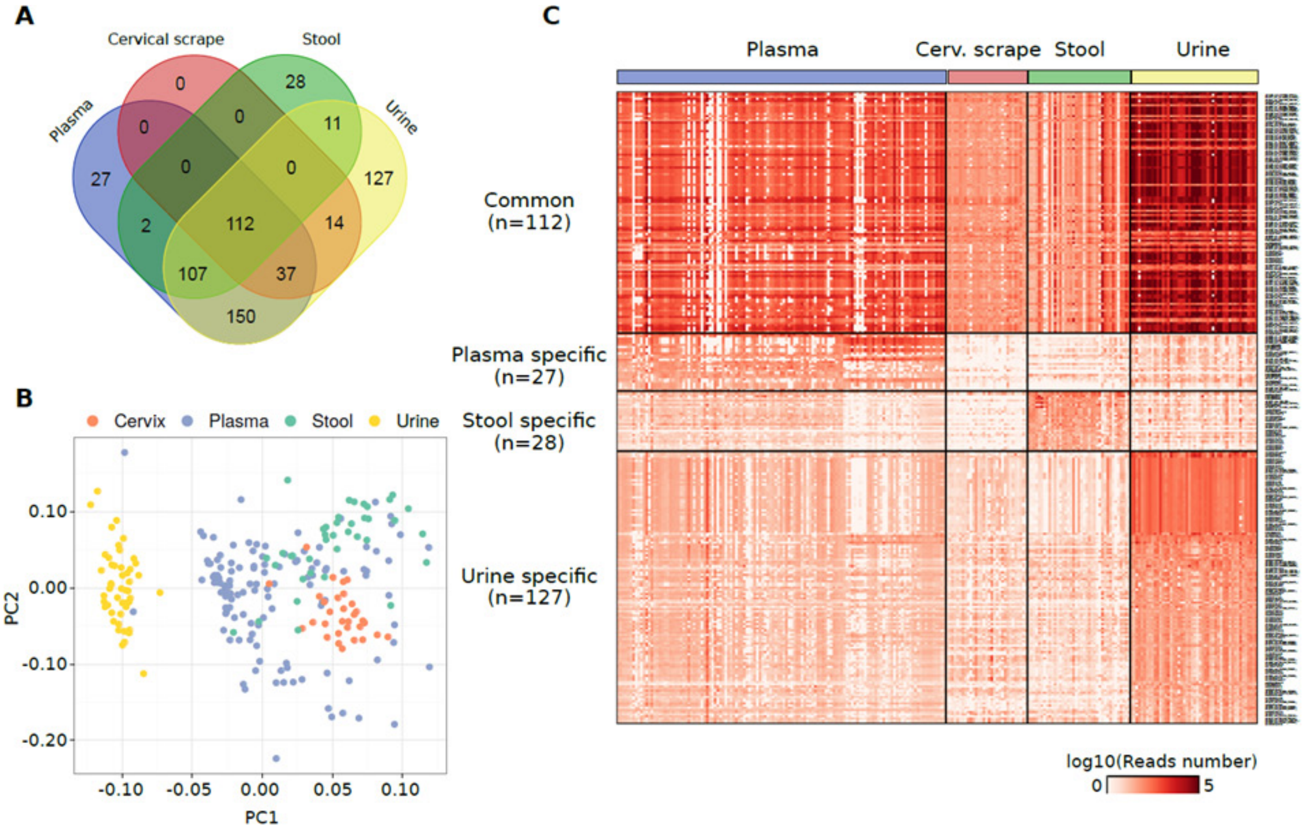


Figure 3: (**A**) Venn diagram reporting the number of non-miRNA sncRNA species detected in different specimens from healthy individuals and their overlap. (**B**) PCA plot showing the small RNA-Seq datasets separation obtained using the non-miRNA RNA species detected in the samples analysed. (**C**) Heat map showing the log10 number of normalized reads supporting the non-miRNA RNA species detected in one specimen only or commonly detected among them.

respectively (Figure 4A, 4B). Considering the isomiRs predicted for the reference miRNAs reported in Figure 4A, all the isomiRs predicted for reference miRNAs in plasma and urine were also identified in these sample types while no isomiRs were predicted for reference miRNAs in stool samples (Supplementary Table 2D). The analysis of *reference* non-miRNA sncRNAs highlighted piR-43137, chr6.tRNA59-IleAAT, and piR-33543 as the candidate sncRNAs for plasma exosome, stool, and urine samples, respectively (Figure 4C, 4D).

To further investigate the *reference* sncRNAs identified, an integrative analysis of public resources was performed (Supplementary Table 4A, 4B). Considering the top 10 reference miRNAs and sncRNAs characterized by the low ratio between MAD and median expression (Figure 4A, 4C), their expression was compared with RNA-Seq data from specimens collected in five independent studies and two databases publicly available. All the 10 most stably expressed miRNAs in plasma exosomes were also detected (average reads >20) in exosome data (analysed individuals, *n* = 40) from [15], plasma samples (*n* = 55) from [6], and venous blood samples data (*n* = 3) from [22]. Six out of 10 stably expressed urine miRNAs were detected in urine RNA (*n* = 4 and *n* = 55 analysed by [22] and [6], respectively). Interestingly, six of the 10 top miRNAs were also detected in samples from kidney (*n* = 11) or bladder (*n* = 2) small RNA-Seq data from DASHR database. The expression of the top 10 miRNAs in stool samples was not confirmed in stool data (*n* = 2) from [22], but two miRNAs were detected in colon samples (*n* = 8) by [23].

Among the reference non-miRNA sncRNAs, piR-62011 was detected as abundant in our plasma exosome data as well as plasma, serum and whole blood data from DASHR. Chr6.tRNA152-ValCAC was detected in our urine set and in small RNA-Seq data from DASHR

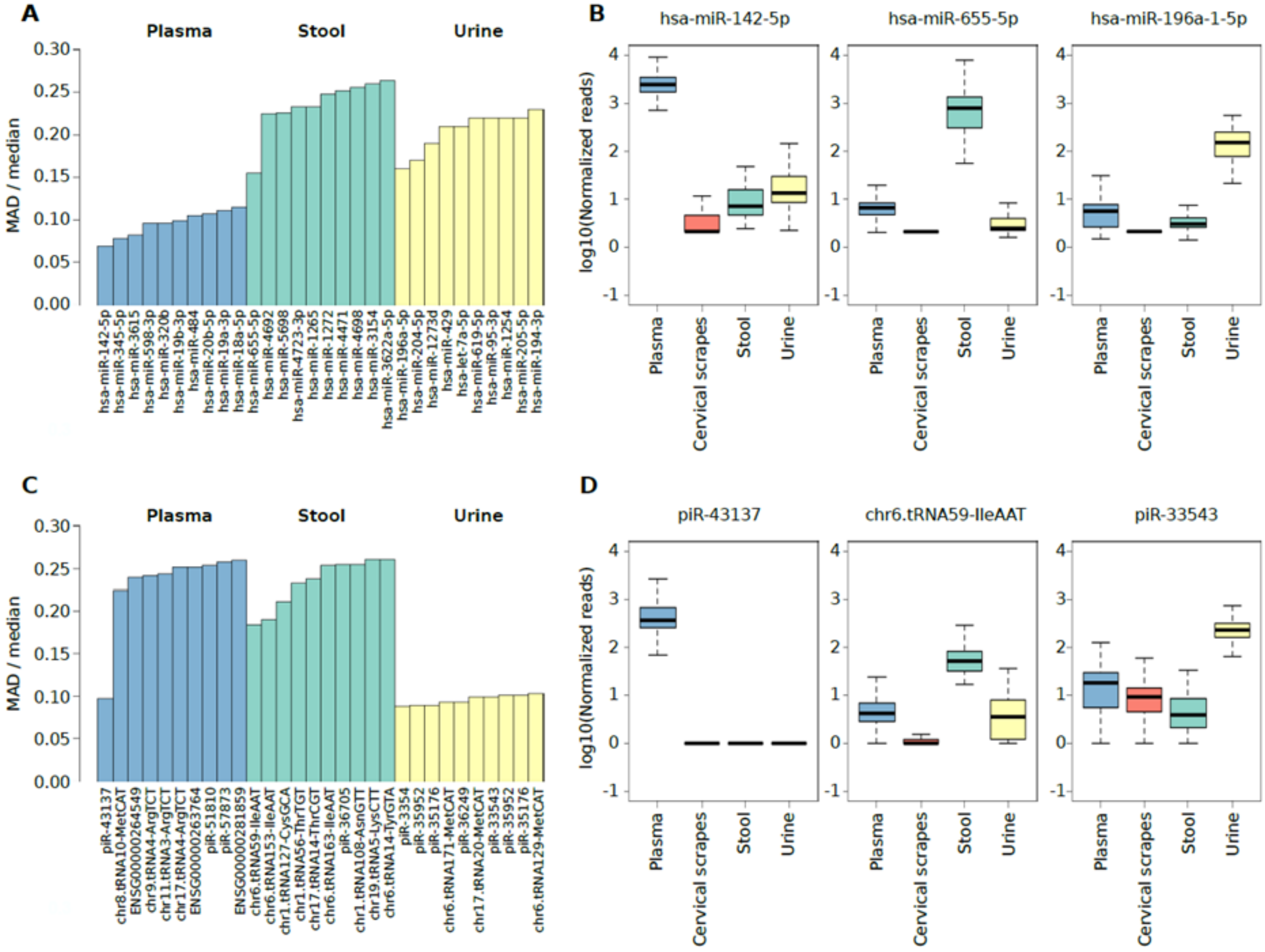


Figure 4: (**A**) Bar plot showing the top 10 miRNAs characterized by the lower ratio between the MAD and the median expression levels in plasma exosome, stool, or urine samples. (**B**) Box plot showing the log10 number of normalized reads supporting miRNAs characterized by the lower ratio between the MAD and the median expression level in plasma exosomes stool, or urine samples. (**C**) Bar plot showing the top 10 non-miRNA sncRNA species characterized by the lower ratio between the MAD and the median expression levels in plasma exosome, stool, or urine samples. (**D**) Box plot showing the log10 number of normalized reads supporting non-miRNA sncRNA species characterized by the lower ratio between the MAD and the median expression level in plasma exosomes stool, or urine samples.

kidney tissues [21]. All the others reference non-miRNA sncRNAs were generally associated with low expression in most of the datasets analysed.

DISCUSSION

The study of the expression patterns of different sncRNAs in a wide spectrum of tissues, along with investigations into the functions of these molecules, is yielding novel insights in the fast-growing field of non-coding RNAs in the normal cell biology and pathogenesis. miRNAs have been extensively studied in the extracellular space but little is still known about the presence of other sncRNAs [15]. As diagnostic and therapeutic procedures move from biopsies in the direction to less invasive methodologies, sncRNAs analysed in different biospecimens represent attractive candidates as biomarkers for complex diseases [12].

In the present study, we investigated expression patterns of sncRNAs in different human biospecimens that could be easily and minimally invasively collected also in the context of screening programs. The data presented hereby were obtained from healthy subjects representing, on average, the steady state in normal conditions of the human organism.

The first analysis was focused on miRNA expression distribution across different investigated specimens. Globally, setting up an arbitrary threshold of median 20 reads, almost 400 miRNAs (out of an average of 1,046 unique miRNAs identified across specimens with at least one read) were detected, with many of them specific to one or few specimen types. A large set of miRNAs was expressed only in plasma exosomes (*n* = 155) while less miRNAs were private of stool or urine and only one of cervical scrapes. Plasma exosomes also shared several miRNAs with other specimens (particularly urine with 109 expressed miRNAs in common). Interestingly, considering the whole set of highly expressed miRNAs, it was possible to accurately group samples of the same biological type independently from the others. This aspect is important in search of specific biomarkers representing an altered status of a tissue in relation to a disease [24]. Conversely, eleven miRNAs presented a similar pattern of expression among all specimens. The most commonly investigated resulted miR-320a whose downregulation is associated with different diseases including cancer [25–30]. The relevance of an ubiquitous high expression of this miRNA related to a healthy status is supported by our findings as well. miR-589-5p, miR-636, and miR-4792 have been also described previously in other studies. miR-589-5p resulted a good inhibitor of MAP3K8 and suppressor of CD90+ cancer stem cells in hepatocellular carcinoma [31]. On the other hand, miR-636 was proposed as a good biomarker for several diseases in a large set of tissues and biofluids such as diabetic kidney disease [32], colorectal cancer [33], and pancreatic cancer [34]. Finally, miR-4792 was found dysregulated in oral submucous fibrosis [35], in nasopharyngeal carcinoma tissues [36] and in uterine leiomyoma [37]. Surprisingly, the rest of the commonly expressed miRNAs were not studied in detail before. Besides being found dysregulated in many studies in relation to different diseases, those miRNAs commonly expressed across different types of samples could be taken into consideration as multi-specimen markers. We have compared our results to those of available datasets on same specimens or anatomically-related tissues [6, 15, 16, 21, 22, 38]. The total number of reads obtained and the proportion of the detected sncRNA species is comparable to other studies previously published with the exception of the study of Yeri and colleagues that included YRNAs [6, 15, 38]. For instance, the high expression of the above mentioned miR-320a and miR-589-5p were also observed in all other datasets.

Notably, in our study, we could compare the co-expression of sncRNAs in plasma exosomes/urine or plasma exosomes/stool collected from the same subjects. Again, in the search of specific markers related to disease, it is important to have an overview on the similarities/differences across different biotypes at an individual level. Apparently, except for very few miRNAs mostly detected in urine/plasma, we could not observe any significant relationship between the expression of same sncRNAs in different biospecimens. This aspect is very important, in the sense that a multi-specimen miRNA panel may be more relevant for accurately describing a disease status, providing different miRNA behaviours across tissues. Similar findings were reported by us in a study on miRNA expression levels in both stool and whole plasma of healthy subjects with different dietary habits. Despite similar associations were observed between miRNA and diet (vegans, vegetarian vs omnivorous) or lifestyle habits, miRNA expression levels were not related between the two different specimens [39].

Since isomiRs have emerged as widely expressed in normal and cancer tissues [40, 41], we further investigated whether they were also detectable in the analysed specimens. As reported in Supplementary Table 2C–2D, many isomiRs were predicted in our datasets particularly in plasma and urine samples. Interestingly, among the 11 miRNAs commonly expressed in all specimens, eight were associated to an isomiR predicted in only one or two types of them.

miRNA profiling by NGS in different specimens in relation to healthy status and pathological conditions is becoming more and more frequent, especially in whole plasma [15]. Less explored is the field of other non-miRNA sncRNAs, although RNA sequencing potentialities, new annotation tools available and an increasing number of studies demonstrating their role in the normal physiology of the organism are appearing [42]. These ‘new’ small RNAs may play an important role in RNA silencing, micro-guarding and cancer [43]. In our study, we have confirmed that small RNA-Seq provide a huge number of reads not mapping to the miRNome in all type of samples analysed, particularly in stool. However, there is still not a consensus on how to comprehensively analyse these RNA molecules. In the present study, we focused on RNA species with a size between 30 to 70 nucleotides, due to the characteristics of the libraries prep kit employed, specific for small RNA sequencing. Considering these criteria, we have obtained a potential list of thousands of RNAs (>30,000) which we have used to filter the remapped reads after their annotation (from DASHR and GENCODE databases). Despite several different sncRNAs identifiable with our thresholds (misc_RNA, Mt_tRNA, piRNA, rRNA, snoRNA, snRNA, sRNA, tRNA), we have mainly identified piRNAs and tRNAs. In urine, we observed the largest number of "private" sncRNAs other than miRNAs (*n* = 127). Cervical scrapes had the less abundant number of these species and none of them was private. In total, 112 sncRNAs resulted expressed in all the biospecimens. Again, plasma exosome and urine samples shared many molecules in common (*n* = 150). Interestingly, as for miRNAs, also for the other sncRNAs, several molecules were characteristics of a single specimen while others were in common. Each body fluid appears to have clear differences in extracellular RNA expression profiles. For example, there appears to be a high proportion of piRNAs in urine samples, when compared with other RNA biotypes. This is quite similar to what observed by Yeri *et al.* [6] which observed an overrepresentation in urine of piRNAs and tRNAs. piRNAs hold great promise as potential biomarkers, owing to their sncRNA features such as small size, stability in biofluids and archival materials, and the variety of detection methods. Moreover, considering there are 10–25 times more piRNA species (20,000–50,000) than miRNAs, the impact of their deregulation is likely at least as relevant. Additionally, piRNA expression patterns have been shown to be deregulated in a variety of cancer types [44–46]. Recently, the study of tRNAs and their role in the regulation of gene expression is revealing new interesting aspects in molecular biology. tRNA-derived small RNAs, named tRNA halves (tiRNAs) and tRNA fragments (tRFs), have been reported to be abundant and their dysregulation to be associated with cancer [43]. Interestingly, we have not identified snoRNAs and other sncRNAs as reported in other studies [6, 47]. Better sncRNA tissue atlases that include more comprehensive profiles of the small RNA species will be necessary for better comparisons.

Expression patterns of miRNAs have been extensively studied but there is still controversy on the best endogenous control(s) to employ as reference in studies by RT-qPCR or microarray, especially when analysing biofluids [24]. An overview of the expression levels of sncRNAs in a large set of biofluids/biospecimen could provide a good base for the research of endogenous controls to be used in case-control studies when searching for sncRNAs as biomarkers of disease [24]. We propose miR-142-5p, miR-655-5p, and miR-196a-1-5p as miRNAs with a high and stable expression in plasma exosome, stool, and urine respectively, while piR-43137, chr6.tRNA59-IleAAT, and piR-33543 as the candidate references among other sncRNAs in plasma exosomes, stool, and urine respectively. miR-142-5p has been found dysregulated in plasma but not in exosomes [47–49] although it has been demonstrated that in rats the activation of the acute stress response modifies its profile in plasma exosomes [50]. miR-655-5p and miR-196a-1-5p have never been studied in stool and urine, except for miR-196a reported to be altered in focal segmental glomerulosclerosis [51]. Considering the top 10 *reference* miRNAs detected in plasma exosomes, stool, or urine sample group, we observed a general coherence between the specificity of isomiR and reference miRNA expression. The only exceptions were two 5′ variants of miR-204. However, these variants were detected by imposing two and three 5′ mismatches on a 14- and 15 nt sub-sequence of miR-204, respectively. The read alignment against such small sequences makes the read assignment less reliable reinforcing the hypothesis that a deeper sequencing depth is required to characterize properly the expression of these miRNA variants.

The biological samples used in the present work are very attractive for the research of non-invasive biomarkers. Blood plasma and urine belong to the group of easily accessible body fluids, and they are among the most frequently used diagnostic material for the development of surrogate cancer biomarkers [52, 53]. From the first work reporting the presence in plasma of miRNAs by Lawrie and colleagues [54], a growing number of studies have evaluated their expression in relation to a wide range of diseases and focused on the biology and features of circulating miRNAs [55]. Circulating miRNAs are considered as a tool employed in the horizontal gene transfer between cells within the tumor or between tumor and host cells: this is a strong biological rationale to use them as a new class of cancer biomarkers. miRNAs and other sncRNAs can be released by the cell by passive leakage into circulation. However, these molecular species can be released in a more active way from the cells by secretion of shedding microvesicles or exosomes containing free sncRNAs or in the form of ribonucleoprotein complexes [56]. Bladder cells are in direct contact with urine making this body fluid an ideal source for the detection of cancer biomarkers. Urine is collected noninvasively, and the procedure is relatively fast and cost-efficient compared with other clinical samples. In addition, sampling can be repeated at different times, and this makes urine an attractive candidate as a screening test for urogenital cancers that needs constant monitoring [53]. Stool has been extensively used as a potential substrate for developing non-invasive molecular screening tests for gastrointestinal diseases including colorectal cancer and for microbiome analyses. There is a rationale for determination of noncoding RNAs expression levels in stool which includes the observations that colonocytes are continuously shed into the faecal stream, with a periodicity of exfoliation roughly every 3–4 days. In addition, sncRNAs are extremely stable, enabling accurate and reproducible detection in the stool without need of special stabilization or logistical requirements. Conventional stool-based screening tests present several limitations including low sensitivity and specificity for advanced adenoma and pre-cancerous lesions. No optimal method has been established yet based on faecal DNA- and mRNA-based testing [57]. The role of diet and other lifestyle factors on miRNA and other sncRNA expression profiles in relation to disease risk is still scarcely explored [58]. Dietary components have been implicated in many pathways involved in diseases, including apoptosis, cell-cycle control, inflammation, and angiogenesis. Those pathways are also regulated by different RNAs [59]. Interestingly, recent discoveries point to a role of faecal miRNAs also introduced by the diet on shaping the human microbiota [60]. Cervical exfoliated cells are widely used in cervical cancer screening, both for HPV testing and Pap test. Recently, their use has been extended to miRNA analyses [61]. These few studies show that the potential application of miRNA detection in cervical exfoliated cells deserves further exploration, also as an additional option for triage of HPV-positive women in population-based screening.

We acknowledge that the present study has some limitations but also several strengths. Among the latter, we can consider the large number of samples sequenced, especially for plasma-exosomes, and the possibility to analyse different biospecimens of the same subjects to understand different/similar patterns according to tissue of origin. To our knowledge, we report the largest description of sncRNA data from plasma-derived exosome, as well as the first investigation of this kind on cervical scrape and stool samples by NGS in healthy subjects. Importantly, the outcomes of our study derive from samples analysed with the same protocols by the same operators and analysed by the same pipeline from raw sequencing data to final results. Other studies usually combine different datasets from different studies.

Among the limitations of our study, we can list that the library preparation is optimized for miRNAs while we have also adapted it for detecting a group of other sncRNAs. Additionally, we could not control analyses considering known potential confounders (age, gender) since not all the samples were provided with this information. Finally, some of the samples were investigated only in subjects of one gender only (i.e., urine in males only).

Small RNA-Seq holds promise for exhaustively analyse miRNAs and other sncRNAs in many different types of specimens, as we demonstrated in our study. These RNA molecules are currently investigated for their potential use as diagnostic/prognostic tools. The high resistance to degradation of sncRNAs makes these molecules particularly attractive for researchers that constantly cope with a wide range of incubations and storage conditions, as well as different origins of samples [62]. However, an optimization and standardization of both the biological and computational procedures to investigate sncRNA expression levels are necessary. Combining molecular aspects with bioinformatics and an epidemiological approach should provide stronger markers to be investigated specifically in particular biospecimens.

MATERIALS AND METHODS

Study participants

All samples included in the study were collected from healthy donors participating to different studies running in our laboratories who donated their blood (for plasma extraction), stool, and /or urine for research purposes [63, 64]. For cervical scrapes, samples were collected in the context of a national screening programme (New Technologies for Cervical Cancer screening (NTCC) study, [65]). All subjects provided written informed consent according to the Helsinki declaration. The design of the study was approved by the local Ethics Committees.

Stool samples (study 1)

In a hospital-based study for colorectal cancer diagnosis, subjects resulting negative to colonoscopy and to any inflammatory disease were included in the present study. For the same individuals, we have collected also plasma samples (n = 39). Naturally evacuated stool samples were collected in special tubes with RNA stabilizing solution, returned at the time of performing colonoscopy and stored at –80°C until RNA extraction.

Urine (study 2)

The study population included men recruited between the years 2008–2012 in the Turin Bladder Cancer Study (TBCS) who donated an aliquot of blood and urine. A full description of controls is available in Pardini *et al.* [66]. For almost all subjects, we have collected also plasma samples (n = 46).

Urine samples from each participant were collected in the morning, stored at 4°C until the processing consisting of centrifugation at 3,000g for 10 min. The urine supernatant aliquots were then transferred in tubes and stored at –80°C until use.

Exosome isolation from plasma

In addition to the subjects described above for whom plasma samples were available (Study 1 and Study 2), we have included also 40 plasma samples collected from healthy blood donors for a Leukaemia study (Study 3).

For all subjects, human plasma samples were obtained from 5–8 ml of blood centrifuged for 10 min at 1000 rpm. Plasma aliquots (about 200–300 µl each) were then stored at –80°C until use. Exosomes were isolated from 200 µl of plasma using the ExoQuick exosome precipitation solution (System Biosciences, Mountain View, CA, USA) according to the manufacturer's instructions with minor modifications. Briefly, the plasma was mixed with 50.4 µl of ExoQuick solution and refrigerated at 4°C overnight (at least 12 h). The mixture was then further centrifuged at 1500 g for 30 min. The exosome pellet was dissolved in 200 µl of nuclease free water; RNA was extracted immediately from the solution.

Cervical scrapes

The study is nested in a large Italian multi-centre randomised controlled trial recruiting women in population-based screening programs that actively invite women aged 25–64 years (NTCC Study, [65]). NTCC recruitment was conducted between 2002 and 2004. In the present study, only samples from HPV negative women were included. Cervical scrape samples have been collected and stored in Specimen Transport Medium (STM), or RNA-later at –80°C until RNA extraction.

RNA extraction and quality control

Total RNA from plasma exosomes was extracted with the miRNeasy plasma/serum mini kit (Qiagen) using the QiaCube extractor (Qiagen). RNA from stool was extracted using the Stool Total RNA Purification Kit (Norgen Biotek Corp). Total RNA from urine was extracted with Urine microRNA Purification kit (Norgen biotek corp), following the manufacturer's standard protocol.

RNA from cervical scrape was extracted from samples stored in STM or RNA-later, using the miRCURY™ RNA Isolation Kit - Cell & Plant (Exiqon) following manufacturer`s protocol.

RNA quality and quantity was verified according to MIQE guidelines (http://miqe.gene-quantification.info/). For all samples, RNA concentration was quantified by Qubit® 2.0 Fluorometer with Qubit® microRNA Assay Kit (Invitrogen).

Library preparation for small RNA-Seq

Small RNA transcripts were converted into barcoded cDNA libraries. Library preparation was performed with the NEBNext Multiplex Small RNA Library Prep Set for Illumina (New England BioLabs Inc., USA). For each library, 6 µL of RNA (min 35 ng) were used in all the experimental procedures as starting material. Each library was prepared with a unique indexed primer so that the libraries could all be pooled into one sequencing lane. Multiplex adaptor ligations, reverse transcription primer hybridization, reverse transcription reaction and PCR amplification were performed according to the protocol for library preparation (Protocol E7330, New England BioLabs Inc., USA). After PCR amplification, the cDNA constructs were purified with the QIAQuick PCR Purification Kit (Qiagen, Germany) following the modifications suggested by the NEBNext Multiplex Small RNA Library Prep Protocol and loaded on the Bioanalyzer 2100 (Agilent, Germany) using the DNA High Sensitivity Kit (Agilent, Germany) according to the manufacturer's protocol. Libraries were pooled together (24plex) and further purified with a gel size selection.

A concluding Bioanalyzer 2100 run with the High Sensitivity DNA Kit (Agilent Technologies, Germany) that allows the analysis of DNA libraries regarding size, purity and concentration completed the workflow of library preparation. The obtained sequence libraries were subjected to the Illumina sequencing pipeline, passing through clonal cluster generation on a single-read flow cell (Illumina Inc., USA) by bridge amplification on the cBot (TruSeq SR Cluster Kit v3-cBOT-HS, Illumina Inc., USA) and 50 cycles sequencing-by-synthesis on the HiSeq 2000 (Illumina Inc., USA) (in collaboration with EMBL, Heidelberg, Germany).

Computational analyses (additional information in Supplementary Material)

Analysis of miRNAs

miRNA data analysis was performed following the optimized workflow proposed in [19]. The obtained FASTQ files from small RNA-seq were quality-checked using FastQC software.

Reads shorter than 14 nucleotides were discarded from the analysis; the remaining reads were clipped from the adapter sequences using Cutadapt software (http://journal.embnet.org/index.php/embnetjournal/article/view/200). The trimmed reads were mapped against the precursor miRNA sequences downloaded from miRBase (Release 21) by the Shrimp algorithm. A matrix of integer values called counting matrix was created.

Since plasma datasets were generated in independent studies and presented a large variability, a SVA [67] was performed to correct the read counts. IsomiR analysis was performed using isomiRID algorithm [68] in default settings. A maximum of three mismatches between reads and reference miRNA sequences was considered for the analysis.

Analysis of other sncRNAs

The set of small RNA-Seq reads not aligned by SHRiMP over miRNA sequences were aligned against human genomic sequence hg38 (GRCh38) using Bowtie2 v2.2.7 in default settings [69]. Reads alignment files were used to quantify the expression of ncRNA annotations from Gencode v24 [70] and DASHR database [21]. The annotations with median reads greater than 20 were selected. Then, read counts were normalized by computing the library size factor [71]. The SVA [67] was performed to correct the read counts of plasma studies.

Bioinformatic tools and data integration

The list and the expression levels of sncRNAs identified in the different specimen types were compared using Venn diagrams and *heatmap.2* R functions. PCA analysis was performed using *prcomp* R function and *autoplot* function from *ggfortify* R package. The contribution of each sncRNA expression level to the classification of specimen type was evaluated using Weka 3.6.12 [72]. miRNA functional enrichment analysis was performed using EnrichR web tool [73] on the list of validated miRNA targets annotated in miRWalk 2.0 database [74].

The set of sncRNAs identified in this study was compared with public lists sncRNAs detected in specimens and tissues from healthy individuals as reported in supplementary materials of target publications and databases.

Abbreviations

isomiR: isoform of miRNAs; miRNA: microRNA; piRNAs: Piwi-interacting RNA; tRNAs: transfer RNA; RT-qPCR: Quantitative reverse transcription PCR; sncRNA: small non-coding RNA; snoRNAs: small nucleolar RNAs; NGS: Next-Generation Sequencing; RNA-Seq: RNA-Sequencing; HPV: Human Papilloma Virus; DASHR: Database of Small Human non-coding RNAs; PCA: principal component analysis; mRNA: messenger RNA; MAD: Median Absolute Deviation; tiRNAs: tRNA halves; tRFs: tRNA fragments; TBCS: Turin Bladder Cancer Study; NTCC: New Technologies for CC screening; STM: Specimen Transport Medium; GRCh38: Genome Reference Consortium Human Build 38; SVA: Surrogate Variable Analysis.

Author contributions

Conception and design of the study: BP, AN, GF, FC; Acquisition, analysis and interpretation of data: BP, AN, GF, FC, MA, FR, RAC, ST, GG; Draft of the manuscript: BP, GF, FC, ST, AN; Critical revision of the manuscript: MA, FR, GG, GR, MAl, NK , GM, PV, RAC.

ACKNOWLEDGMENTS

The authors are very thankful to all volunteers that participated to the present study. We wish to thank EMBL Genomics Core Facilities (EMBL, http://www.genecore.embl.de) for sequencing of our libraries.

FUNDING

This work was supported by the Italian Institute for Genomic Medicine (IIGM) and Compagnia di San Paolo Torino, Italy (to B.P., A.N., P.V. and G.M.), by Fondazione Umberto Veronesi "Post-doctoral fellowship Year 2014, 2015, 2016 and 2017" (B.P. and S.T. recipients), by the Italian Association on Cancer Research (IG2013 N.14119 to G.R and A.N.), by Lega Italiana per La Lotta contro i Tumori (to F.C. and A.N.) and by the Grant Agency of the Czech Republic (17-16857S to A.N.).

CONFLICTS OF INTEREST

The authors declare no competing interests.

REFERENCES

1. Cech TR, Steitz JA. The noncoding RNA revolution-trashing old rules to forge new ones. Cell. 2014; 157:77–94. https://doi.org/10.1016/j.cell.2014.03.008.
2. Bracken CP, Scott HS, Goodall GJ. A network-biology perspective of microRNA function and dysfunction in cancer. Nat Rev Genet. 2016; 17:719–32. https://doi.org/10.1038/nrg.2016.134.
3. Cheng Y, Tan N, Yang J, Liu X, Cao X, He P, Dong X, Qin S, Zhang C. A translational study of circulating cell-free microRNA-1 in acute myocardial infarction. Clin Sci (Lond). 2010; 119:87–95. https://doi.org/10.1042/CS20090645.
4. Calin GA, Ferracin M, Cimmino A, Di Leva G, Shimizu M, Wojcik SE, Iorio MV, Visone R, Sever NI, Fabbri M, Iuliano R, Palumbo T, Pichiorri F, et al. A MicroRNA signature associated with prognosis and progression in chronic lymphocytic leukemia. N Engl J Med. 2005; 353:1793–801. https://doi.org/10.1056/NEJMoa050995.
5. Esteller M. Non-coding RNAs in human disease. Nat Rev Genet. 2011; 12:861–74. https://doi.org/10.1038/nrg3074.
6. Yeri A, Courtright A, Reiman R, Carlson E, Beecroft T, Janss A, Siniard A, Richholt R, Balak C, Rozowsky J, Kitchen R, Hutchins E, Winarta J, et al. Total Extracellular Small RNA Profiles from Plasma, Saliva, and Urine of Healthy Subjects. Sci Rep. 2017; 7:44061. https://doi.org/10.1038/srep44061.
7. Buschmann D, Haberberger A, Kirchner B, Spornraft M, Riedmaier I, Schelling G, Pfaffl MW. Toward reliable biomarker signatures in the age of liquid biopsies - how to standardize the small RNA-Seq workflow. Nucleic Acids Res. 2016; 44:5995–6018. https://doi.org/10.1093/nar/gkw545.
8. Weber JA, Baxter DH, Zhang S, Huang DY, Huang KH, Lee MJ, Galas DJ, Wang K. The microRNA spectrum in 12 body fluids. Clin Chem. 2010; 56:1733–41. https://doi.org/10.1373/clinchem.2010.147405.
9. Pritchard CC, Cheng HH, Tewari M. MicroRNA profiling: approaches and considerations. Nat Rev Genet. 2012; 13:358–69. https://doi.org/10.1038/nrg3198.
10. Accerbi M, Schmidt SA, De Paoli E, Park S, Jeong DH, Green PJ. Methods for isolation of total RNA to recover miRNAs and other small RNAs from diverse

species. Methods Mol Biol. 2010; 592:31–50. https://doi.org/10.1007/978-1-60327-005-2_3.

11. Lopez JP, Diallo A, Cruceanu C, Fiori LM, Laboissiere S, Guillet I, Fontaine J, Ragoussis J, Benes V, Turecki G, Ernst C. Biomarker discovery: quantification of microRNAs and other small non-coding RNAs using next generation sequencing. BMC Med Genomics. 2015; 8:35. https://doi.org/10.1186/s12920-015-0109-x.
12. Witwer KW, Halushka MK. Toward the promise of microRNAs - Enhancing reproducibility and rigor in microRNA research. RNA Biol. 2016; 13:1103–16. https://doi.org/10.1080/15476286.2016.1236172.
13. Margue C, Reinsbach S, Philippidou D, Beaume N, Walters C, Schneider JG, Nashan D, Behrmann I, Kreis S. Comparison of a healthy miRNome with melanoma patient miRNomes: are microRNAs suitable serum biomarkers for cancer? Oncotarget. 2015; 6:12110–27. https://doi.org/10.18632/oncotarget.3661.
14. Yuan T, Huang X, Woodcock M, Du M, Dittmar R, Wang Y, Tsai S, Kohli M, Boardman L, Patel T, Wang L. Plasma extracellular RNA profiles in healthy and cancer patients. Sci Rep. 2016; 6:19413. https://doi.org/10.1038/srep19413.
15. Freedman JE, Gerstein M, Mick E, Rozowsky J, Levy D, Kitchen R, Das S, Shah R, Danielson K, Beaulieu L, Navarro FC, Wang Y, Galeev TR, et al. Diverse human extracellular RNAs are widely detected in human plasma. Nat Commun. 2016; 7:11106. https://doi.org/10.1038/ncomms11106.
16. Ben-Dov IZ, Whalen VM, Goilav B, Max KE, Tuschl T. Cell and Microvesicle Urine microRNA Deep Sequencing Profiles from Healthy Individuals: Observations with Potential Impact on Biomarker Studies. PLoS One. 2016; 11:e0147249. https://doi.org/10.1371/journal.pone.0147249.
17. Fehlmann T, Ludwig N, Backes C, Meese E, Keller A. Distribution of microRNA biomarker candidates in solid tissues and body fluids. RNA Biol. 2016; 13:1084–88. https://doi.org/10.1080/15476286.2016.1234658.
18. Bahn JH, Zhang Q, Li F, Chan TM, Lin X, Kim Y, Wong DT, Xiao X. The landscape of microRNA, Piwi-interacting RNA, and circular RNA in human saliva. Clin Chem. 2015; 61:221–30. https://doi.org/10.1373/clinchem.2014.230433.
19. Cordero F, Beccuti M, Arigoni M, Donatelli S, Calogero RA. Optimizing a massive parallel sequencing workflow for quantitative miRNA expression analysis. PLoS One. 2012; 7:e31630. https://doi.org/10.1371/journal.pone.0031630.
20. Wright JC, Mudge J, Weisser H, Barzine MP, Gonzalez JM, Brazma A, Choudhary JS, Harrow J. Improving GENCODE reference gene annotation using a high-stringency proteogenomics workflow. Nat Commun. 2016; 7:11778. https://doi.org/10.1038/ncomms11778.
21. Leung YY, Kuksa PP, Amlie-Wolf A, Valladares O, Ungar LH, Kannan S, Gregory BD, Wang LS. DASHR: database of small human noncoding RNAs. Nucleic Acids Res. 2016; 44:D216–22. https://doi.org/10.1093/nar/gkv1188.
22. Seashols-Williams S, Lewis C, Calloway C, Peace N, Harrison A, Hayes-Nash C, Fleming S, Wu Q, Zehner ZE. High-throughput miRNA sequencing and identification of biomarkers for forensically relevant biological fluids. Electrophoresis. 2016; 37:2780–88. https://doi.org/10.1002/elps.201600258.
23. Hamfjord J, Stangeland AM, Hughes T, Skrede ML, Tveit KM, Ikdahl T, Kure EH. Differential expression of miRNAs in colorectal cancer: comparison of paired tumor tissue and adjacent normal mucosa using high-throughput sequencing. PLoS One. 2012; 7:e34150. https://doi.org/10.1371/journal.pone.0034150.
24. Hardwick SA, Deveson IW, Mercer TR. Reference standards for next-generation sequencing. Nat Rev Genet. 2017; 18:473–84. https://doi.org/10.1038/nrg.2017.44.
25. Xie F, Yuan Y, Xie L, Ran P, Xiang X, Huang Q, Qi G, Guo X, Xiao C, Zheng S. miRNA-320a inhibits tumor proliferation and invasion by targeting c-Myc in human hepatocellular carcinoma. Onco Targets Ther. 2017; 10:885–94. https://doi.org/10.2147/OTT.S122992.
26. Lv Q, Hu JX, Li YJ, Xie N, Song DD, Zhao W, Yan YF, Li BS, Wang PY, Xie SY. MiR-320a effectively suppresses lung adenocarcinoma cell proliferation and metastasis by regulating STAT3 signals. Cancer Biol Ther. 2017; 18:142–51. https://doi.org/10.1080/15384047.2017.1281497.
27. Yu J, Wang L, Yang H, Ding D, Zhang L, Wang J, Chen Q, Zou Q, Jin Y, Liu X. Rab14 Suppression Mediated by MiR-320a Inhibits Cell Proliferation, Migration and Invasion in Breast Cancer. J Cancer. 2016; 7:2317–26. https://doi.org/10.7150/jca.15737.
28. Cordes F, Brückner M, Lenz P, Veltman K, Glauben R, Siegmund B, Hengst K, Schmidt MA, Cichon C, Bettenworth D. MicroRNA-320a Strengthens Intestinal Barrier Function and Follows the Course of Experimental Colitis. Inflamm Bowel Dis. 2016; 22:2341–55. https://doi.org/10.1097/MIB.0000000000000917.
29. Tadano T, Kakuta Y, Hamada S, Shimodaira Y, Kuroha M, Kawakami Y, Kimura T, Shiga H, Endo K, Masamune A, Takahashi S, Kinouchi Y, Shimosegawa T. MicroRNA-320 family is downregulated in colorectal adenoma and affects tumor proliferation by targeting CDK6. World J Gastrointest Oncol. 2016; 8:532–42. https://doi.org/10.4251/wjgo.v8.i7.532.
30. Xishan Z, Ziying L, Jing D, Gang L. MicroRNA-320a acts as a tumor suppressor by targeting BCR/ABL oncogene in chronic myeloid leukemia. Sci Rep. 2015; 5:12460. https://doi.org/10.1038/srep12460.
31. Zhang X, Jiang P, Shuai L, Chen K, Li Z, Zhang Y, Jiang Y, Li X. miR-589-5p inhibits MAP3K8 and suppresses CD90+ cancer stem cells in hepatocellular carcinoma. J Exp Clin Cancer Res. 2016; 35:176.
32. Eissa S, Matboli M, Aboushahba R, Bekhet MM, Soliman Y. Urinary exosomal microRNA panel unravels novel biomarkers for diagnosis of type 2 diabetic kidney disease.

J Diabetes Complications. 2016; 30:1585–92. https://doi.org/10.1016/j.jdiacomp.2016.07.012.

33. Slattery ML, Herrick JS, Mullany LE, Wolff E, Hoffman MD, Pellatt DF, Stevens JR, Wolff RK. Colorectal tumor molecular phenotype and miRNA: expression profiles and prognosis. Mod Pathol. 2016; 29:915–27.

34. Schultz NA, Dehlendorff C, Jensen BV, Bjerregaard JK, Nielsen KR, Bojesen SE, Calatayud D, Nielsen SE, Yilmaz M, Holländer NH, Andersen KK, Johansen JS. MicroRNA biomarkers in whole blood for detection of pancreatic cancer. JAMA. 2014; 311:392–404. https://doi.org/10.1001/jama.2013.284664.

35. Chickooree D, Zhu K, Ram V, Wu HJ, He ZJ, Zhang S. A preliminary microarray assay of the miRNA expression signatures in buccal mucosa of oral submucous fibrosis patients. J Oral Pathol Med. 2016; 45:691–697. https://doi.org/10.1111/jop.12431.

36. Li Y, Chen X. miR-4792 inhibits epithelial-mesenchymal transition and invasion in nasopharyngeal carcinoma by targeting FOXC1. Biochem Biophys Res Commun. 2015; 468:863–69. https://doi.org/10.1016/j.bbrc.2015.11.045.

37. Georgieva B, Milev I, Minkov I, Dimitrova I, Bradford AP, Baev V. Characterization of the uterine leiomyoma microRNAome by deep sequencing. Genomics. 2012; 99:275–81. https://doi.org/10.1016/j.ygeno.2012.03.003.

38. Ainsztein AM, Brooks PJ, Dugan VG, Ganguly A, Guo M, Howcroft TK, Kelley CA, Kuo LS, Labosky PA, Lenzi R, McKie GA, Mohla S, Procaccini D, et al, and The NIH Extracellular RNA Communication Consortium. The NIH Extracellular RNA Communication Consortium. J Extracell Vesicles. 2015; 4:27493. https://doi.org/10.3402/jev.v4.27493.

39. Tarallo S, Pardini B, Mancuso G, Rosa F, Di Gaetano C, Rosina F, Vineis P, Naccarati A. MicroRNA expression in relation to different dietary habits: a comparison in stool and plasma samples. Mutagenesis. 2014; 29:385–91. https://doi.org/10.1093/mutage/geu028.

40. Telonis AG, Magee R, Loher P, Chervoneva I, Londin E, Rigoutsos I. Knowledge about the presence or absence of miRNA isoforms (isomiRs) can successfully discriminate amongst 32 TCGA cancer types. Nucleic Acids Res. 2017; 45:2973–85. https://doi.org/10.1093/nar/gkx082.

41. McCall MN, Kim MS, Adil M, Patil AH, Lu Y, Mitchell CJ, Leal-Rojas P, Xu J, Kumar M, Dawson VL, Dawson TM, Baras AS, Rosenberg AZ, et al. Toward the human cellular microRNAome. Genome Res. 2017; 27:1769–81. https://doi.org/10.1101/gr.222067.117.

42. Ng KW, Anderson C, Marshall EA, Minatel BC, Enfield KS, Saprunoff HL, Lam WL, Martinez VD. Piwi-interacting RNAs in cancer: emerging functions and clinical utility. Mol Cancer. 2016; 15:5. https://doi.org/10.1186/s12943-016-0491-9.

43. Green D, Fraser WD, Dalmay T. Transfer RNA-derived small RNAs in the cancer transcriptome. Pflugers Arch. 2016; 468:1041–47. https://doi.org/10.1007/s00424-016-1822-9.

44. Cui L, Lou Y, Zhang X, Zhou H, Deng H, Song H, Yu X, Xiao B, Wang W, Guo J. Detection of circulating tumor cells in peripheral blood from patients with gastric cancer using piRNAs as markers. Clin Biochem. 2011; 44:1050–57. https://doi.org/10.1016/j.clinbiochem.2011.06.004.

45. Zhang H, Ren Y, Xu H, Pang D, Duan C, Liu C. The expression of stem cell protein Piwil2 and piR-932 in breast cancer. Surg Oncol. 2013; 22:217–23. https://doi.org/10.1016/j.suronc.2013.07.001.

46. Chu H, Hui G, Yuan L, Shi D, Wang Y, Du M, Zhong D, Ma L, Tong N, Qin C, Yin C, Zhang Z, Wang M. Identification of novel piRNAs in bladder cancer. Cancer Lett. 2015; 356:561–67. https://doi.org/10.1016/j.canlet.2014.10.004.

47. Brenu EW, Ashton KJ, Batovska J, Staines DR, Marshall-Gradisnik SM. High-throughput sequencing of plasma microRNA in chronic fatigue syndrome/myalgic encephalomyelitis. PLoS One. 2014; 9:e102783. https://doi.org/10.1371/journal.pone.0102783.

48. Sørensen SS, Nygaard AB, Christensen T. miRNA expression profiles in cerebrospinal fluid and blood of patients with Alzheimer's disease and other types of dementia - an exploratory study. Transl Neurodegener. 2016; 5:6. https://doi.org/10.1186/s40035-016-0053-5.

49. Huang S, Deng Q, Feng J, Zhang X, Dai X, Li L, Yang B, Wu T, Cheng J. Polycyclic Aromatic Hydrocarbons-Associated MicroRNAs and Heart Rate Variability in Coke Oven Workers. J Occup Environ Med. 2016; 58:e24–31. https://doi.org/10.1097/JOM.0000000000000564.

50. Beninson LA, Brown PN, Loughridge AB, Saludes JP, Maslanik T, Hills AK, Woodworth T, Craig W, Yin H, Fleshner M. Acute stressor exposure modifies plasma exosome-associated heat shock protein 72 (Hsp72) and microRNA (miR-142-5p and miR-203). PLoS One. 2014; 9:e108748. https://doi.org/10.1371/journal.pone.0108748.

51. Zhang W, Zhang C, Chen H, Li L, Tu Y, Liu C, Shi S, Zen K, Liu Z. Evaluation of microRNAs miR-196a, miR-30a-5P, and miR-490 as biomarkers of disease activity among patients with FSGS. Clin J Am Soc Nephrol. 2014; 9:1545–52. https://doi.org/10.2215/CJN.11561113.

52. Gaedcke J, Grade M, Camps J, Sokilde R, Kaczkowski B, Schetter AJ, Difilippantonio MJ, Harris CC, Ghadimi BM, Moller S, Beissbarth T, Ried T, Litman T. The rectal cancer microRNAome—microRNA expression in rectal cancer and matched normal mucosa. Clin Cancer Res. 2012; 18:4919–30.

53. Matullo G, Naccarati A, Pardini B. MicroRNA expression profiling in bladder cancer: the challenge of next-generation sequencing in tissues and biofluids. Int J Cancer. 2016; 138:2334–45. https://doi.org/10.1002/ijc.29895.

54. Lawrie CH, Gal S, Dunlop HM, Pushkaran B, Liggins AP, Pulford K, Banham AH, Pezzella F, Boultwood J, Wainscoat JS, Hatton CS, Harris AL. Detection of elevated levels of tumour-associated microRNAs in serum of patients with diffuse large B-cell lymphoma.

Br J Haematol. 2008; 141:672–75. https://doi.org/10.1111/j.1365-2141.2008.07077.x.

55. Vodicka P, Pardini B, Vymetalkova V, Naccarati A. Polymorphisms in Non-coding RNA Genes and Their Targets Sites as Risk Factors of Sporadic Colorectal Cancer. Adv Exp Med Biol. 2016; 937:123–49. https://doi.org/10.1007/978-3-319-42059-2_7.

56. Redova M, Sana J, Slaby O. Circulating miRNAs as new blood-based biomarkers for solid cancers. Future Oncol. 2013; 9:387–402. https://doi.org/10.2217/fon.12.192.

57. Di Lena M, Travaglio E, Altomare DF. New strategies for colorectal cancer screening. World J Gastroenterol. 2013; 19:1855–60. https://doi.org/10.3748/wjg.v19.i12.1855.

58. Vilaprinyo E, Forné C, Carles M, Sala M, Pla R, Castells X, Domingo L, Rue M, and Interval Cancer (INCA) Study Group. Cost-effectiveness and harm-benefit analyses of risk-based screening strategies for breast cancer. PLoS One. 2014; 9:e86858. https://doi.org/10.1371/journal.pone.0086858 .

59. Ross SA, Davis CD. MicroRNA, nutrition, and cancer prevention. Adv Nutr. 2011; 2:472–85. https://doi.org/10.3945/an.111.001206.

60. Liu S, da Cunha AP, Rezende RM, Cialic R, Wei Z, Bry L, Comstock LE, Gandhi R, Weiner HL. The Host Shapes the Gut Microbiota via Fecal MicroRNA. Cell Host Microbe. 2016; 19:32–43. https://doi.org/10.1016/j.chom.2015.12.005.

61. He Y, Lin J, Ding Y, Liu G, Luo Y, Huang M, Xu C, Kim TK, Etheridge A, Lin M, Kong D, Wang K. A systematic study on dysregulated microRNAs in cervical cancer development. Int J Cancer. 2016; 138:1312–27. https://doi.org/10.1002/ijc.29618.

62. Mitchell PS, Parkin RK, Kroh EM, Fritz BR, Wyman SK, Pogosova-Agadjanyan EL, Peterson A, Noteboom J, O'Briant KC, Allen A, Lin DW, Urban N, Drescher CW, et al. Circulating microRNAs as stable blood-based markers for cancer detection. Proc Natl Acad Sci USA. 2008; 105:10513–18. https://doi.org/10.1073/pnas.0804549105.

63. Critelli R, Fasanelli F, Oderda M, Polidoro S, Assumma MB, Viberti C, Preto M, Gontero P, Cucchiarale G, Lurkin I, Zwarthoff EC, Vineis P, Sacerdote C, et al. Detection of multiple mutations in urinary exfoliated cells from male bladder cancer patients at diagnosis and during follow-up. Oncotarget. 2016; 7:67435–48. https://doi.org/10.18632/oncotarget.11883.

64. Russo A, Modica F, Guarrera S, Fiorito G, Pardini B, Viberti C, Allione A, Critelli R, Bosio A, Casetta G, Cucchiarale G, Destefanis P, Gontero P, et al. Shorter leukocyte telomere length is independently associated with poor survival in patients with bladder cancer. Cancer Epidemiol Biomarkers Prev. 2014; 23:2439–46. https://doi.org/10.1158/1055-9965.EPI-14-0228.

65. Bergeron C, Giorgi-Rossi P, Cas F, Schiboni ML, Ghiringhello B, Dalla Palma P, Minucci D, Rosso S, Zorzi M, Naldoni C, Segnan N, Confortini M, Ronco G. Informed cytology for triaging HPV-positive women: substudy nested in the NTCC randomized controlled trial. J Natl Cancer Inst. 2015; 107:107. https://doi.org/10.1093/jnci/dju423.

66. Pardini B, Viberti C, Naccarati A, Allione A, Oderda M, Critelli R, Preto M, Zijno A, Cucchiarale G, Gontero P, Vineis P, Sacerdote C, Matullo G. Increased micronucleus frequency in peripheral blood lymphocytes predicts the risk of bladder cancer. Br J Cancer. 2017; 116:202–10. https://doi.org/10.1038/bjc.2016.411.

67. Leek JT, Johnson WE, Parker HS, Jaffe AE, Storey JD. The sva package for removing batch effects and other unwanted variation in high-throughput experiments. Bioinformatics. 2012; 28:882–83. https://doi.org/10.1093/bioinformatics/bts034.

68. de Oliveira LF, Christoff AP, Margis R. isomiRID: a framework to identify microRNA isoforms. Bioinformatics. 2013; 29:2521–23. https://doi.org/10.1093/bioinformatics/btt424.

69. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. Nat Methods. 2012; 9:357–59. https://doi.org/10.1038/nmeth.1923.

70. Harrow J, Frankish A, Gonzalez JM, Tapanari E, Diekhans M, Kokocinski F, Aken BL, Barrell D, Zadissa A, Searle S, Barnes I, Bignell A, Boychenko V, et al. GENCODE: the reference human genome annotation for The ENCODE Project. Genome Res. 2012; 22:1760–74. https://doi.org/10.1101/gr.135350.111.

71. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. 2014; 15:550. https://doi.org/10.1186/s13059-014-0550-8.

72. Frank E, Hall M, Trigg L, Holmes G, Witten IH. Data mining in bioinformatics using Weka. Bioinformatics. 2004; 20:2479–81. https://doi.org/10.1093/bioinformatics/bth261.

73. Kuleshov MV, Jones MR, Rouillard AD, Fernandez NF, Duan Q, Wang Z, Koplev S, Jenkins SL, Jagodnik KM, Lachmann A, McDermott MG, Monteiro CD, Gundersen GW, Ma'ayan A. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. Nucleic Acids Res. 2016; 44:W90–7. https://doi.org/10.1093/nar/gkw377.

74. Dweep H, Gretz N. miRWalk2.0: a comprehensive atlas of microRNA-target interactions. Nat Methods. 2015; 12:697. https://doi.org/10.1038/nmeth.3485.