

Original Paper

Omics Pipe: A Community-based Framework for Reproducible Multi-Omics Data Analysis

Kathleen M. Fisch^{1*}, Tobias Meißner^{1†}, Louis Gioia¹, Jean-Christophe Ducom², Tristan M. Carland³, Salvatore Loguerzio¹ & Andrew I. Su^{1*}

¹Department of Molecular and Experimental Medicine, The Scripps Research Institute, 10550 North Torrey Pines Road, La Jolla, CA, USA 92037

²The Scripps Research Institute, 10550 North Torrey Pines Road, La Jolla, CA, USA 92037

³Department of Human Biology, J. Craig Venter Institute, 4120 Capricorn Lane, La Jolla, CA, USA 92037

Associate Editor: Dr. John Hancock

ABSTRACT

Motivation: Omics Pipe (<http://sulab.scripps.edu/omicspipe>) is a computational framework that automates multi-omics data analysis pipelines on high performance compute clusters and in the cloud. It supports best practice published pipelines for RNA-seq, miRNA-seq, Exome-seq, Whole Genome sequencing, ChIP-seq analyses and automatic processing of data from The Cancer Genome Atlas (TCGA). Omics Pipe provides researchers with a tool for reproducible, open source and extensible next generation sequencing analysis. The goal of Omics Pipe is to democratize NGS analysis by dramatically increasing the accessibility and reproducibility of best practice computational pipelines, which will enable researchers to generate biologically meaningful and interpretable results.

Results: Using Omics Pipe, we analyzed 100 TCGA breast invasive carcinoma paired tumor-normal data sets based on the latest UCSC hg19 RefSeq annotation. Omics Pipe automatically downloaded and processed the desired TCGA samples on a high throughput compute cluster to produce a results report for each sample. We aggregated the individual sample results and compared them to the analysis in the original publications. This comparison revealed high overlap between the analyses, as well as novel findings due to the use of updated annotations and methods.

Availability and Implementation: Source code for Omics Pipe is freely available on the web (https://bitbucket.org/sulab/omics_pipe). Omics Pipe is distributed as a standalone Python package for installation (https://pypi.python.org/pypi/omics_pipe) and as an Amazon Machine Image (AMI) in Amazon Web Services (AWS) Elastic Compute Cloud (EC2) that contains all necessary third-party software dependencies and databases (https://pythonhosted.org/omics_pipe/AWS_installation.html).

Contact: asu@scripps.edu or kfisch@ucsd.edu

Supplementary Information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Next generation sequencing (NGS) has presented researchers with the opportunity to collect large amounts of sequencing data (Mardis, 2011), which has accelerated the pace of genomic research with applications to personalized medicine and diagnostics (Nocq et al., 2013). These technologies have resulted in the development of a large number of computational tools and analysis pipelines, necessitating the creation of best practices and reproducible integrative analysis frameworks (Nekrutenko and Taylor, 2012). Several automated pipelines have been developed to tie together individual software tools such as bcbio-nextgen (<https://github.com/chapmanb/bcbio-nextgen>), Galaxy (Goecks et al., 2010) and others (Golosova et al., 2014; Nocq et al., 2013). In addition, several groups have published step-by-step directions for well-established best practices in NGS analysis. Examples include the Broad Institute's best practices for variant calling using the Genome Analysis Toolkit (GATK) (McKenna et al., 2010) and ENCODE's standardized data-processing guidelines (ENCODE Project Consortium, 2012).

Despite the establishment of best practices and analysis pipelines, there exists a need for a platform that provides researchers with an analysis tool that can be easily understood and reproduced by other researchers in a variety of computational environments (Nekrutenko and Taylor, 2012). Galaxy, a widely known platform for performing reproducible computational analyses, has been designed to make computational analyses accessible to non-programmers (Goecks et al., 2010). It allows the user to pipeline together various software tools pre-wrapped by Galaxy without the need to learn the implementation details of any single tool in a web-based interface. Other biologist-oriented NGS software tools include Unipro UGENE NGS pipelines (Golosova et al., 2014) and the workflow management system Taverna (Wolstencroft et al., 2013). While these tools are extremely useful for biologists and other non-programmers, more advanced users may have a need for a tool that supports programmatic access to the individual tools, is easily extensible and is reproducible.

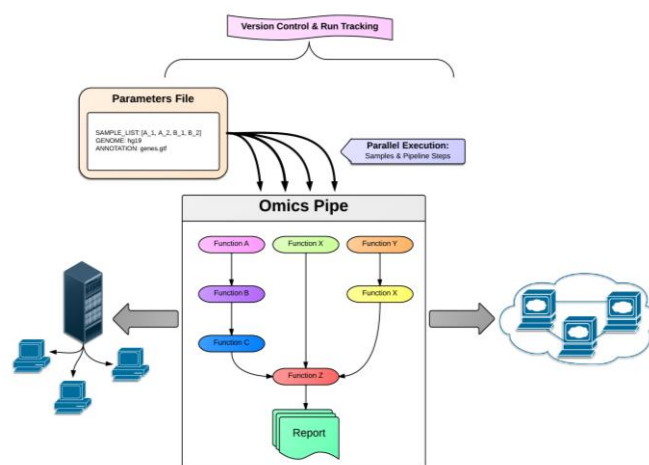
*To whom correspondence should be addressed.

†These authors contributed equally to this work.

Several tools exist to allow developers to pipeline together functions, such as Bpipe (Sadedin et al., 2012), Snakemake (Koster and Rahmann, 2012), and Ruffus (Goodstadt, 2010), although developers are still required to develop their own pipelines using these frameworks. A complete analysis pipeline, Bcbio-nextgen (<https://github.com/chapmanb/bcbio-nextgen>), is designed for bioinformaticians to implement high throughput optimized NGS analysis pipelines and requires familiarity with command-line programming. The focus of bcbio-nextgen is on variant calling using a variety of software tools that have been performance optimized for use in bcbio-nextgen. This makes bcbio-nextgen a powerful tool for implementing variant calling pipelines; however, customizing and extending a bcbio-nextgen pipeline requires extensive programming knowledge. In addition, there are several other pipelining tools, although many focus only on a single NGS platform, require computational expertise, require commercial licenses and/or are poorly documented, necessitating the need for an open-source computational tool that provides researchers with a reproducible framework to democratize NGS analyses (Nekrutenko and Taylor, 2012).

To address this need, we developed Omics Pipe (<http://sulab.scripps.edu/omicspipe>), an open-source, modular computational platform that provides a community-curated framework for automating best practice multi-omics data analysis pipelines with built-in version control for reproducibility (Figure 1). It currently supports several best practice next-generation sequencing pipelines (Figure 2). A non-programmer with basic unix command-line programming experience can easily execute the supported pipelines within Omics Pipe, although the target users for Omics Pipe are computational biologists and bioinformaticians that require full programmatic access to the individual software tools and parameters. The Omics Pipe framework is modular, which allows researchers to easily and efficiently add new analysis tools with scripts in the form of Python modules that can then be used to assemble a new analysis pipeline. Detailed tutorials (http://pythonhosted.org/omics_pipe/), documentation and source code are hosted in an open source repository that will allow community contribution to the source code as well as transparency for reproducibility and accuracy (https://bitbucket.org/sulab/omics_pipe).

One advantage of a robust and simple to use framework is the ability to easily reanalyze existing data sets using the most recent algorithms and annotations. To illustrate this point, we used Omics Pipe to reanalyze a subset of the breast invasive carcinoma RNA-seq dataset (N = 100) paired tumor-normal samples generated by the TCGA Research Network (<http://cancergenome.nih.gov/>) using the count-based differential expression analysis best practice protocol (Anders et al., 2013) and updated UCSC RefSeq annotations (V57). Omics Pipe automatically downloaded the desired TCGA samples and ran the selected pipeline on a high throughput compute cluster. We performed paired differential expression analysis, signaling pathway impact analysis (Tarca et al., 2009) and consensus clustering analysis (Wilkerson and Hayes, 2010). We aggregated the individual sample results to compare the results of our analysis with the original dataset, which revealed high overlap between the analyses, as well as novel findings due to the use of updated annotations and methods. In conclusion, Omics pipe enables researchers to analyze next generation sequencing data with little development overhead to provide reproducible, open source and



extensible use of established multi-omics analysis methods by providing researchers with a community-curated best practice NGS analysis framework.

Fig. 1. Schematic diagram of Omics Pipe demonstrating the parallel execution of pipelined tasks and samples. Omics Pipe requires a parameter file in YAML format, and can be run on a local compute cluster or in the cloud. Each run of Omics Pipe is logged with the version and run information for reproducibility

2 IMPLEMENTATION

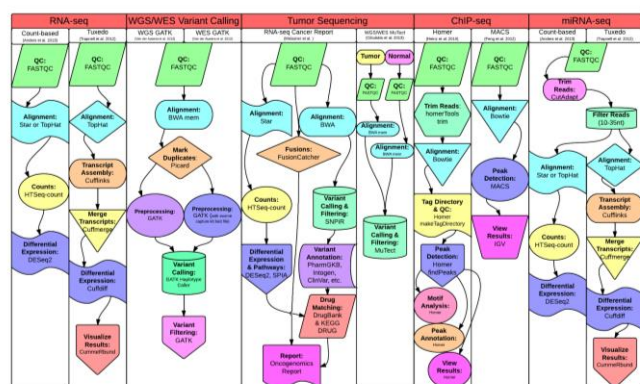
Omics Pipe is a Python package that creates a framework for assembling scripts into an automated, version controlled, parallelized pipeline for bioinformatics analyses. Omics Pipe uses the Python package Ruffus (Goodstadt, 2010) for running the pipeline steps, Sumatra (Davison, 2012) for version control and run tracking, and Python DRMAA (<https://github.com/drmaa-python>) for distributed computing. Omics Pipe is distributed as a standalone Python package for installation on a local cluster. Third party software dependencies and reference databases must be available on the local cluster for Omics Pipe to run. Omics Pipe is also distributed as an Amazon Machine Image (AMI) in Amazon Web Services (AWS) Elastic Compute Cloud (EC2) that contains all necessary third-party software dependencies and databases. The AWS distribution of Omics Pipe runs on MIT's StarCluster (<http://star.mit.edu/>). A Docker container (<https://www.docker.com/>) is provided to configure and boot up StarCluster with the preconfigured Omics Pipe AMI.

Users can either choose from a predefined set of supported pipelines, or specify the path to a custom Python script containing a custom pipeline. Users have full control to define relevant parameters for running the pipeline, including the command line options for each tool and other customizable settings, through a parameter file in YAML format. All of the parameters have default values to enable the user to run the supported pipelines with minimal start up time. More advanced users can customize every option possible from each of the pipelined tools.

Omics Pipe can be extended by the user to create custom pipelines from built-in modules and by creating simple module wrappers for new tools. It is language agnostic, so existing scripts written in any programming language can be included as an Omics

Pipe module. Omics Pipe executes the scripts on the cluster or in the cloud using DRMAA to allocate resources and manage job execution. Omics Pipe checks that each job in a pipeline finishes successfully and creates a flag file upon successful completion, allowing the user to rerun only incomplete steps in the pipeline. Ruffus (Goodstadt, 2010) provides functionality for parallel execution of pipeline steps. Each time Omics Pipe is executed, Sumatra (Davison, 2012) creates a database entry to log the specifics of the run, including the parameters, input files, output files and software versions for version control and run tracking.

Omics Pipe currently supports six published best practice pipelines -- two RNA sequencing (RNA-seq) pipelines (Anders et al., 2013; Trapnell et al., 2012), variant calling from whole exome sequencing (WES) and whole genome sequencing (WGS) based on GATK (McKenna et al., 2010), and two ChIP-seq pipelines (Feng et al., 2012; Heinz et al., 2010). It also includes custom RNA-seq pipelines for personalized cancer genomic medicine reporting



(Meißner et al., 2014) and analysis of The Cancer Genome Atlas (TCGA) datasets (Cancer Genome Atlas, 2012) (Figure 2). The steps in each method have been adapted exactly as described in the associated publications, allowing the user to easily execute these methods on their own datasets. The command-line options for each tool in each pipeline are exposed to the user in the parameters file.

Fig. 2. Pre-built best practice pipelines and the third party software tools supported by Omics Pipe. Users can easily create custom pipelines from the existing modules and they can create new modules supporting additional third party software tools.

3 METHODS

To demonstrate its utility for efficiently processing samples using best practice pipelines, we used Omics Pipe to reanalyze 100 paired tumor/normal samples from 50 patients in the TCGA breast invasive carcinoma dataset. We automatically downloaded the raw RNA-seq fastq files and processed the files using the count-based differential expression analysis best practice protocol (Anders et al., 2013) to quantify gene expression.

Briefly, sequencing reads were aligned to the human genome (hg19) using the STAR aligner (Dobin et al., 2012). Gene expression quantification was performed at the exon level using the htseq-count function within the Python HTSeq analysis package (Anders et al., 2014) with UCSC RefSeq hg19 annotation (Release 57). The 50% most variable genes were used in the differential expression analysis after TMM normalization

(Robinson and Oshlack, 2010). Differential gene expression was performed using a paired design matrix with the Bioconductor package edgeR (Robinson et al., 2010). Genes with a False Discovery Rate (FDR) < 0.01 and $\log_2(\text{FoldChange}) > |2|$ were considered differentially expressed.

For comparison, we also downloaded the raw count files generated from the TCGA UNC V2 RNA-seq Workflow for the same 100 samples. These counts were generated using the UNC V2 RNA-seq Workflow and were based on the UCSC RefSeq hg19 Generic Annotation File from June 2011. We performed differential expression analysis of these raw counts as described above.

3.1 Identification of Novel Genes, Pathways and Clustering in TCGA Breast Invasive Carcinoma

We compared the differentially expressed genes in the reanalysis of the TCGA dataset using Omics Pipe to the raw counts originally produced by TCGA to assess the utility of rerunning previous analyses with updated gene annotations and algorithms. We updated the gene identifiers provided with the original raw count data using the Bioconductor package mygene (Wu et al., 2014) and we extracted newly annotated differentially expressed genes identified in the reanalyzed dataset.

We identified significantly dysregulated pathways in each gene set with the Bioconductor packages SPIA (Tarca et al., 2009) and Graphite (Sales et al., 2012) based on the Biocarta, KEGG, NCI and Reactome databases. We performed this analysis once on each dataset, using the differentially expressed genes in each dataset as input, and setting the background genes to all genes included in the differential expression analysis for each dataset.

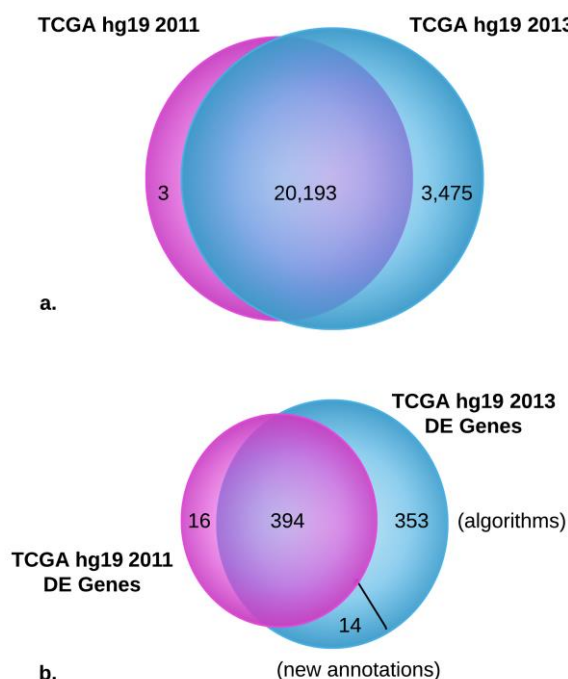
We identified relationships among the samples based on gene expression data in each dataset using the Bioconductor package ConsensusClusterPlus (Wilkerson and Hayes, 2010) with 80% resampling from 2 to 20 clusters and 1,000 iterations of hierarchical clustering based on a Pearson correlation distance metric. We compared the differentially expressed gene sets, pathways and clusters between the previously published results and the current analysis using updated annotations to identify novel differentially expressed genes and pathways relevant to breast cancer.

4 RESULTS

We reanalyzed 100 samples in 80 hours using 10 Dell Poweredge M600 blades with two 2.66 GHz Intel quad core E5430 XEON-EMT processors and 16GB of ECC DDR2 memory (6-8 hours per sample). Runtime on other systems will vary depending on the hardware specs and the number of nodes used. The updated UCSC RefSeq V57 annotations contained 3,475 additional genes compared to the UCSC RefSeq General Annotation Format from 2011 used to originally analyze the TCGA data (Figure 3a). The reanalysis of the TCGA breast invasive carcinoma samples using Omics Pipe revealed 761 differentially expressed (DE) genes compared to the original TCGA analysis, which resulted in 410 DE genes (Supplementary Tables 1 & 2). There were 394 DE genes shared between the two analyses (Figure 3b). In the reanalyzed dataset, 367 DE genes were unique, 14 of which were due to new

annotations. This result is expected, as different algorithms and annotations are used in this study compared to the original analysis. One of the newly annotated DE genes, DSCAM-AS1, was upregulated 256x in tumor versus normal samples and has been implicated in the malignant progression of breast carcinomas by an estrogen-independent mechanism (Liu et al., 2002).

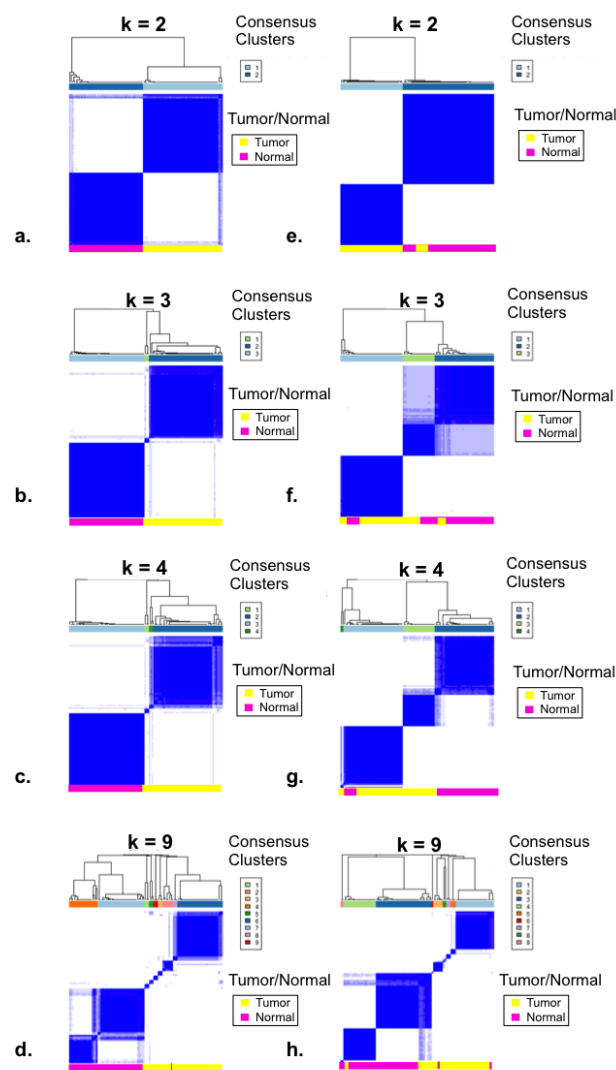
Consensus clustering was performed to determine the number and membership of possible clusters within the dataset. Consensus clustering of the original TCGA counts resulted in four clusters, with each cluster containing both tumor and normal samples (Figures 4 & 5). Consensus clustering of the reanalyzed counts resulted in 10 clusters, with tumor and normal samples clustering separately, with the exception of one normal sample clustering with two tumor samples in Cluster 7 (Figures 4 & 5). These results indicate that improved quantification of genes common to both datasets and the addition of the 3,475 genes in the new annotation provide additional information to improve the separation of tumor and normal samples. Twenty significantly dysregulated pathways were identified from the DE genes from the original TCGA counts, and 29 significantly dysregulated pathways were identified in the reanalyzed dataset. Eleven newly identified pathways were primarily related to RNA polymerase activity



(Supplementary Tables 3 & 4), dysregulation of which has been implicated in mediating malignant transformation in cancer (Bywater et al., 2013). The reanalysis of the TCGA data using a best practice pipeline and updated annotations demonstrates the utility of Omics Pipe as a tool for conducting reproducible NGS analyses that can lead to novel biological insights.

Fig. 3. Comparison of the number of genes annotated in two different UCSC RefSeq releases and the number of differentially expressed (DE) genes identified by different algorithms and annotations (a) Venn diagram of the number of genes annotated in the UCSC RefSeq hg19 2011 Generic Annotation File and the UCSC RefSeq hg19 2013 annotation (Release 57)

(b) Venn diagram of the comparison of the number of DE genes identified between raw counts generated with the TCGA UNC V2 RNA-seq Workflow using the UCSC RefSeq hg19 2011 Generic Annotation File and raw counts generated with the count-based pipeline in Omics Pipe using the



UCSC RefSeq hg19 2013 annotation (Release 57).

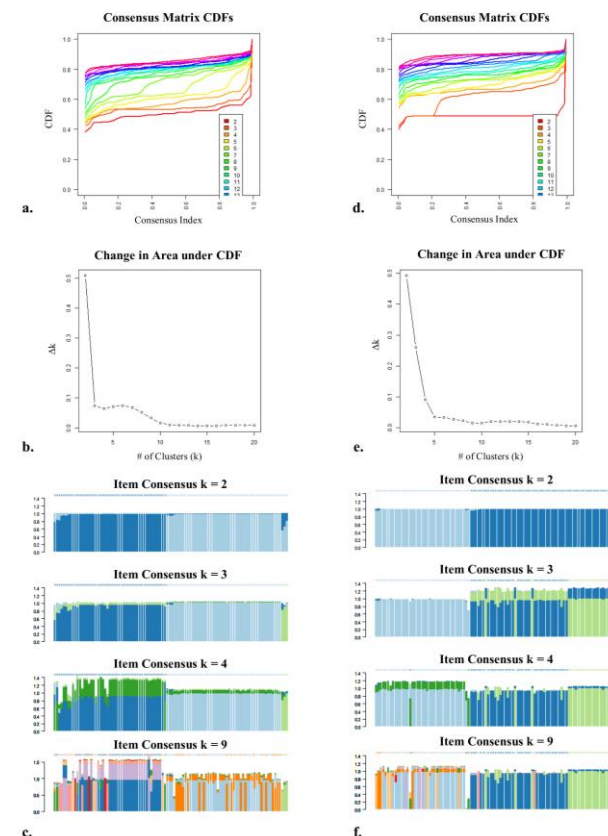
Fig. 4. Consensus clustering analysis of the TCGA breast invasive carcinoma paired tumor-normal samples performed with the reanalyzed count data (a)-(d) and the original raw counts downloaded from TCGA (e)-(h) for cluster sizes of $k=2$, $k=3$, $k=4$ and $k=10$. The blue and white heat map displays sample consensus.

5 DISCUSSION

Omics Pipe is an automated and reproducible community-based framework that can be used to efficiently analyze newly generated data, to reanalyze publicly available data, and to serve as a framework for community-curated NGS analysis pipelines. It currently supports several best practice pipelines for RNA-seq,

WES, WGS and ChIP-seq. This list of pipelines will continue to be updated, and we invite the broader community to participate in the development of Omics Pipe through our open source code repository. Pull requests for new components and new pipelines will be promptly reviewed. Future development of Omics Pipe will

Funding: This work was supported by the National Center for Advancing Translational Sciences (Grant UL1TR001114), the National Cancer Institute (Grant CA92577), the National Institute on Alcohol Abuse and Alcoholism (Grants AA007456, AA013525), the National Institute on Drug Abuse (Grant DA030976), and by a fellowship from the National Foundation for Cancer Research.



include hosting community-curated pipelines and modules. In addition, the built-in version control system allows for the reproducibility of analyses performed within the Omics Pipe framework, which is important as new versions of software tools and annotations are released. It can be easily extended as new tools become available, and it can be implemented on a local machine, a computer cluster or the cloud. The goal of Omics Pipe is to democratize NGS analysis by dramatically increasing the accessibility and reproducibility of best practice computational pipelines, which will enable researchers to generate biologically meaningful and interpretable results.

Fig. 5. Measurements of consensus for different cluster sizes (k) from the consensus clustering analysis on the reanalyzed (a)-(c) and original counts (d)-(f) from the TCGA paired tumor-normal breast invasive carcinoma samples. The empirical cumulative distribution (CDF) plots (a) and (d) indicate at which k the shape of the curve approaches the ideal step function. Plots (b) and (e) depict the area under the two CDF curves. Item consensus plots (c) and (f) demonstrate the mean consensus of each sample with all other samples in a particular cluster (represented by color).

ACKNOWLEDGEMENTS

REFERENCES

- Anders, S., McCarthy, D.J., Chen, Y., Okoniewski, M., Smyth, G.K., Huber, W. and Robinson, M.D. (2013). Count-based differential expression analysis of RNA sequencing data using R and Bioconductor. *Nature protocols*, 8(9):1765-1786.
- Anders, S., Pyl, P.T. and Huber, W. (2014). HTSeq -- a Python framework to work with high-throughput sequencing data. *Bioinformatics*.
- Bywater, M.J., Pearson, R.B., McArthur, G.A. and Hannan, R.D. (2013). Dysregulation of the basal RNA polymerase transcription apparatus in cancer. *Nature Reviews Cancer*, 13:299-314.
- Cancer Genome Atlas, N. (2012). Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61-70.
- Davison, A. (2012). Automated Capture of Experiment Context for Easier Reproducibility in Computational Research. *Computing in Science & Engineering*, 14(4):48-56.
- Dobin, A., Davis, C., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M. and Gingeras, T.R. (2012). STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, 29(11):15-21.
- ENCODE Project Consortium. (2012). An Integrated Encyclopedia of DNA Elements in the Human Genome. *Nature*, 489(7414):57-74.
- Feng, J., Liu, T., Qin, B., Zhang, Y. and Liu, X.S. (2012). Identifying ChIP-seq enrichment using MACS. *Nature protocols*, 7(9):1728-1740.
- Goecks, J., Nekrutenko, A., Taylor, J. and Galaxy Team, T. (2010). Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome biology*, 11(R86):1-13.
- Golosova, O., Henderson, R., Vaskin, Y., Gabrielian, A., Grekhov, G., Nagarajan, V., Oler, A., Quinones, M., Hurt, D., Fursov, M. and Huyen, Y. (2014). Unipro UGENE NGS pipelines and components for variant calling, RNA-seq and ChIP-seq data analyses. *PeerJ*, 2(e644):1-15.
- Goodstadt, L. (2010). Ruffus: A lightweight Python library for computational pipelines. *Bioinformatics*, 26(21):2778-2779.
- Heinz, S., Benner, C., Spann, N., Bertolino, E., Lin, Y., Laslo, P., Cheng, J.X., Murre, C., Singh, H. and Glass, C. (2010). Simple Combinations of Lineage-Determining Transcription Factors Prime cis-Regulatory Elements Required for Macrophage and B Cell Identities. *Molecular Cell*, 38(4):576-589.
- Koster, J. and Rahmann, S. (2012). Snakemake - A scalable bioinformatics workflow engine. *Bioinformatics*, 28(19):2520-2522.
- Liu, D., Rudland, P.S., Sibson, D.R. and Barraclough, R. (2002). Identification of mRNAs differentially-expressed between benign and malignant breast tumour cells. *British Journal of Cancer*, 87(4):423-431.
- Mardis, E.R. (2011). A decade's perspective on DNA sequencing technology. *Nature*, 470:198-203.
- McKenna, A., Hanna, M., Banks, E., Sivachenko, A., Cibulskis, K., Kernysky, A.M., Garimella, K.V., Altshuler, D., Gabriel, S., Daly, M. and DePristo, M.A. (2010). The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome research*, 20(9):1297-1303.
- Meißner, T., Fisch, K.M., Gioia, L. and Su, A.I. (2014). OncoRep: An n-of-1 reporting tool to support genome-guided treatment for breast cancer patients using RNA-sequencing. *bioRxiv*.
- Nekrutenko, A. and Taylor, J. (2012). Next-generation sequencing data interpretation: enhancing reproducibility and accessibility. *Nature reviews. Genetics*, 13(9):667-672.
- Nocq, J., Celton, M., Gendron, P., Lemieux, S. and Wilhelm, B.T. (2013). Harnessing virtual machines to simplify next-generation DNA sequencing analysis. *Bioinformatics*, 29(17):2075-2083.

-
- Robinson, M.D., McCarthy, D.J. and Smyth, G.K. (2010). edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139-140.
- Robinson, M.D. and Oshlack, A. (2010). A scaling normalization method for differential expression analysis of RNA-seq data. *Genome biology*, 11(R25).
- Sadedin, S., Pope, B. and Oshlack, A. (2012). Bpipe: A tool for running and managing bioinformatics pipelines. *Bioinformatics*, 28(11):1525-1526.
- Sales, G., Calura, E., Cavalieri, D. and Romualdi, C. (2012). graphite - a Bioconductor package to convert pathway topology to gene network. *BMC bioinformatics*, 13:20.
- Tarca, A.L., Draghici, S., Khatri, P., Hassan, S.S., Mittal, P., Kim, J.S., Kim, C.J., Kusanovic, J.P. and Romero, R. (2009). A novel signaling pathway impact analysis. *Bioinformatics*, 25(1):75-82.
- Trapnell, C., Roberts, A., Goff, L., Pertea, G., Kim, D., Kelley, D.R., Pimentel, H., Salzberg, S.L., Rinn, J.L. and Pachter, L. (2012). Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nature protocols*, 7(3):562-578.
- Wilkerson, M.D. and Hayes, D.N. (2010). ConsensusClusterPlus: a class discovery tool with confidence assessments and item tracking. *Bioinformatics*, 26(12):1572-1573.
- Wolstencroft, K., Haines, R., Fellows, D., Williams, A., Withers, D., Owen, S., Soiland-Reyes, S., Dunlop, I., Nenadic, A., Fisher, P., Bhagat, J., Belhajjame, K., Bacall, F., Hardisty, A., Nieva de la Hidalga, A., Balcazar Vargas, M., Sufi, S. and Goble, C. (2013). The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud. *Nucleic acids research*, 41:W557-W561.
- Wu, C.W., Mark, A. and Su, A.I. (2014). MyGene.info: gene annotation query as a service. *bioRxiv*.