

---

# Table of Contents

Preface..... xiii

---

## Part I. Ideology: Data Skills for Robust and Reproducible Bioinformatics

**1. How to Learn Bioinformatics..... 1**

- Why Bioinformatics? Biology's Growing Data 1
- Learning Data Skills to Learn Bioinformatics 4
- New Challenges for Reproducible and Robust Research 5
- Reproducible Research 6
- Robust Research and the Golden Rule of Bioinformatics 8
- Adopting Robust and Reproducible Practices Will Make Your Life Easier, Too 9
- Recommendations for Robust Research 10
  - Pay Attention to Experimental Design 10
  - Write Code for Humans, Write Data for Computers 11
  - Let Your Computer Do the Work For You 12
  - Make Assertions and Be Loud, in Code and in Your Methods 12
  - Test Code, or Better Yet, Let Code Test Code 13
  - Use Existing Libraries Whenever Possible 14
  - Treat Data as Read-Only 14
  - Spend Time Developing Frequently Used Scripts into Tools 15
  - Let Data Prove That It's High Quality 15
- Recommendations for Reproducible Research 16
  - Release Your Code and Data 16
  - Document Everything 16
  - Make Figures and Statistics the Results of Scripts 17
  - Use Code as Documentation 17
- Continually Improving Your Bioinformatics Data Skills 17

---

---

## Part II. Prerequisites: Essential Skills for Getting Started with a Bioinformatics Project

<b>2. Setting Up and Managing a Bioinformatics Project.....</b>	<b>21</b>
Project Directories and Directory Structures	21
Project Documentation	24
Use Directories to Divide Up Your Project into Subprojects	26
Organizing Data to Automate File Processing Tasks	26
Markdown for Project Notebooks	31
Markdown Formatting Basics	31
Using Pandoc to Render Markdown to HTML	35
<b>3. Remedial Unix Shell.....</b>	<b>37</b>
Why Do We Use Unix in Bioinformatics? Modularity and the Unix Philosophy	37
Working with Streams and Redirection	41
Redirecting Standard Out to a File	41
Redirecting Standard Error	43
Using Standard Input Redirection	45
The Almighty Unix Pipe: Speed and Beauty in One	45
Pipes in Action: Creating Simple Programs with Grep and Pipes	47
Combining Pipes and Redirection	48
Even More Redirection: A tee in Your Pipe	49
Managing and Interacting with Processes	50
Background Processes	50
Killing Processes	51
Exit Status: How to Programmatically Tell Whether Your Command Worked	52
Command Substitution	54
<b>4. Working with Remote Machines.....</b>	<b>57</b>
Connecting to Remote Machines with SSH	57
Quick Authentication with SSH Keys	59
Maintaining Long-Running Jobs with nohup and tmux	61
nohup	61
Working with Remote Machines Through Tmux	61
Installing and Configuring Tmux	62
Creating, Detaching, and Attaching Tmux Sessions	62
Working with Tmux Windows	64

<b>5. Git for Scientists.....</b>	<b>67</b>
Why Git Is Necessary in Bioinformatics Projects	68
Git Allows You to Keep Snapshots of Your Project	68
Git Helps You Keep Track of Important Changes to Code	69
Git Helps Keep Software Organized and Available After People Leave	69
Installing Git	70
Basic Git: Creating Repositories, Tracking Files, and Staging and Committing Changes	70
Git Setup: Telling Git Who You Are	70
git init and git clone: Creating Repositories	70
Tracking Files in Git: git add and git status Part I	72
Staging Files in Git: git add and git status Part II	73
git commit: Taking a Snapshot of Your Project	76
Seeing File Differences: git diff	77
Seeing Your Commit History: git log	79
Moving and Removing Files: git mv and git rm	80
Telling Git What to Ignore: .gitignore	81
Undoing a Stage: git reset	83
Collaborating with Git: Git Remotes, git push, and git pull	83
Creating a Shared Central Repository with GitHub	86
Authenticating with Git Remotes	87
Connecting with Git Remotes: git remote	87
Pushing Commits to a Remote Repository with git push	88
Pulling Commits from a Remote Repository with git pull	89
Working with Your Collaborators: Pushing and Pulling	90
Merge Conflicts	92
More GitHub Workflows: Forking and Pull Requests	97
Using Git to Make Life Easier: Working with Past Commits	97
Getting Files from the Past: git checkout	97
Stashing Your Changes: git stash	99
More git diff: Comparing Commits and Files	100
Undoing and Editing Commits: git commit --amend	102
Working with Branches	102
Creating and Working with Branches: git branch and git checkout	103
Merging Branches: git merge	105
Branches and Remotes	106
Continuing Your Git Education	108
 <b>6. Bioinformatics Data.....</b>	 <b>109</b>
Retrieving Bioinformatics Data	110
Downloading Data with wget and curl	110
Rsync and Secure Copy (scp)	113

Data Integrity	114
SHA and MD5 Checksums	115
Looking at Differences Between Data	116
Compressing Data and Working with Compressed Data	118
gzip	119
Working with Gzipped Compressed Files	120
Case Study: Reproducibly Downloading Data	120

---

## Part III. Practice: Bioinformatics Data Skills

<b>7. Unix Data Tools.....</b>	<b>125</b>
Unix Data Tools and the Unix One-Liner Approach: Lessons from Programming Pearls	125
When to Use the Unix Pipeline Approach and How to Use It Safely	127
Inspecting and Manipulating Text Data with Unix Tools	128
Inspecting Data with Head and Tail	129
less	131
Plain-Text Data Summary Information with wc, ls, and awk	134
Working with Column Data with cut and Columns	138
Formatting Tabular Data with column	139
The All-Powerful Grep	140
Decoding Plain-Text Data: hexdump	145
Sorting Plain-Text Data with Sort	147
Finding Unique Values in Uniq	152
Join	155
Text Processing with Awk	157
Bioawk: An Awk for Biological Formats	163
Stream Editing with Sed	165
Advanced Shell Tricks	169
Subshells	169
Named Pipes and Process Substitution	171
The Unix Philosophy Revisited	173
<b>8. A Rapid Introduction to the R Language.....</b>	<b>175</b>
Getting Started with R and RStudio	176
R Language Basics	178
Simple Calculations in R, Calling Functions, and Getting Help in R	178
Variables and Assignment	182
Vectors, Vectorization, and Indexing	183
Working with and Visualizing Data in R	193
Loading Data into R	194

Exploring and Transforming Dataframes	199
Exploring Data Through Slicing and Dicing: Subsetting Dataframes	203
Exploring Data Visually with ggplot2 I: Scatterplots and Densities	207
Exploring Data Visually with ggplot2 II: Smoothing	213
Binning Data with cut() and Bar Plots with ggplot2	215
Merging and Combining Data: Matching Vectors and Merging Dataframes	219
Using ggplot2 Facets	224
More R Data Structures: Lists	228
Writing and Applying Functions to Lists with lapply() and sapply()	231
Working with the Split-Apply-Combine Pattern	239
Exploring Dataframes with dplyr	243
Working with Strings	248
Developing Workflows with R Scripts	253
Control Flow: if, for, and while	253
Working with R Scripts	254
Workflows for Loading and Combining Multiple Files	257
Exporting Data	260
Further R Directions and Resources	261
<b>9. Working with Range Data.....</b>	<b>263</b>
A Crash Course in Genomic Ranges and Coordinate Systems	264
An Interactive Introduction to Range Data with GenomicRanges	269
Installing and Working with Bioconductor Packages	269
Storing Generic Ranges with IRanges	270
Basic Range Operations: Arithmetic, Transformations, and Set Operations	275
Finding Overlapping Ranges	281
Finding Nearest Ranges and Calculating Distance	290
Run Length Encoding and Views	292
Storing Genomic Ranges with GenomicRanges	299
Grouping Data with GRangesList	303
Working with Annotation Data: GenomicFeatures and rtracklayer	308
Retrieving Promoter Regions: Flank and Promoters	314
Retrieving Promoter Sequence: Connection GenomicRanges with Sequence Data	316
Getting Intergenic and Intronic Regions: Gaps, Reduce, and Setdiffs in Practice	319
Finding and Working with Overlapping Ranges	324
Calculating Coverage of GRanges Objects	328
Working with Ranges Data on the Command Line with BEDTools	329
Computing Overlaps with BEDTools Intersect	330
BEDTools Slop and Flank	333
Coverage with BEDTools	335

Other BEDTools Subcommands and pybedtools	336
<b>10. Working with Sequence Data.....</b>	<b>339</b>
The FASTA Format	339
The FASTQ Format	341
Nucleotide Codes	343
Base Qualities	344
Example: Inspecting and Trimming Low-Quality Bases	346
A FASTA/FASTQ Parsing Example: Counting Nucleotides	349
Indexed FASTA Files	352
<b>11. Working with Alignment Data.....</b>	<b>355</b>
Getting to Know Alignment Formats: SAM and BAM	356
The SAM Header	356
The SAM Alignment Section	359
Bitwise Flags	360
CIGAR Strings	363
Mapping Qualities	365
Command-Line Tools for Working with Alignments in the SAM Format	365
Using samtools view to Convert between SAM and BAM	365
Samtools Sort and Index	367
Extracting and Filtering Alignments with samtools view	368
Visualizing Alignments with samtools tvview and the Integrated Genomics Viewer	372
Pileups with samtools pileup, Variant Calling, and Base Alignment Quality	378
Creating Your Own SAM/BAM Processing Tools with Pysam	384
Opening BAM Files, Fetching Alignments from a Region, and Iterating Across Reads	384
Extracting SAM/BAM Header Information from an AlignmentFile Object	387
Working with AlignedSegment Objects	388
Writing a Program to Record Alignment Statistics	391
Additional Pysam Features and Other SAM/BAM APIs	394
<b>12. Bioinformatics Shell Scripting, Writing Pipelines, and Parallelizing Tasks.....</b>	<b>395</b>
Basic Bash Scripting	396
Writing and Running Robust Bash Scripts	396
Variables and Command Arguments	398
Conditionals in a Bash Script: if Statements	401
Processing Files with Bash Using for Loops and Globbing	405
Automating File-Processing with find and xargs	411
Using find and xargs	411
Finding Files with find	412

find's Expressions	413
find's -exec: Running Commands on find's Results	415
xargs: A Unix Powertool	416
Using xargs with Replacement Strings to Apply Commands to Files	418
xargs and Parallelization	419
Make and Makefiles: Another Option for Pipelines	421
<b>13. Out-of-Memory Approaches: Tabix and SQLite.....</b>	<b>425</b>
Fast Access to Indexed Tab-Delimited Files with BGZF and Tabix	425
Compressing Files for Tabix with Bgzip	426
Indexing Files with Tabix	427
Using Tabix	427
Introducing Relational Databases Through SQLite	428
When to Use Relational Databases in Bioinformatics	429
Installing SQLite	431
Exploring SQLite Databases with the Command-Line Interface	431
Querying Out Data: The Almighty SELECT Command	434
SQLite Functions	441
SQLite Aggregate Functions	442
Subqueries	447
Organizing Relational Databases and Joins	448
Writing to Databases	455
Dropping Tables and Deleting Databases	458
Interacting with SQLite from Python	459
Dumping Databases	465
<b>14. Conclusion.....</b>	<b>467</b>
Where to Go From Here?	468
<b>Glossary.....</b>	<b>471</b>
<b>Bibliography.....</b>	<b>479</b>
<b>Index.....</b>	<b>483</b>