Contents

Introduction

Multidimensional Data in Biomedical Research

As modern biosciences advance, researchers increasingly encounter datasets that are influenced by a variety of independent variables, such as time, dosage, and environmental conditions. These variables introduce multidimensional complexity into datasets, challenging traditional analysis methods. For instance, cell adhesion studies, which are crucial for understanding cellular interactions and cancer metastasis, often require analyses across multiple time points and varying adhesion molecule concentrations, demonstrating a time-dependent variability that significantly impacts biological interpretations (Rebl et al., 2010; McKay et al., 1997; Bolado-Carrancio et al., 2020).

Multidimensional data encompass datasets where multiple *independent variables* (here reffered to as *factors*) can influence one *dependent variables* (*outcomes*) (Krzywinski & Savig, 2013). In biomedicine, dependent variables are often continuous (intervals or ratios), whereas independent variables are often categorical (ordinal or nominal), respectively. Categorical variables comprise discrete values called categories or *levels*, which are assigned to experimental conditions or measurement modalities, for example the factor *'time'* could comprise three levels: *'0 h'*, *'24 h'*, and *'48 h'*. Such setups are attractive, because they are compatible with common hypothesis tests, such as ANOVA etc. (Motulsky, 2018): If the levels of one factor are associated with a different outcome, that factor is considered to have an influence on the dependent variable. Multiple factors address multiple hypotheses, including the influence from each individual factor, but also potential interactions between factors. This makes it crucial to design analysis strategies that can reveal the true structure and value of the data (Krzywinski & Savig, 2013).

A primary example of multidimensional data is multiplex RT-qPCR, where the expression levels of various genes are measured across different samples under varying conditions (Bustin, 2014). Here, the dependent variable is typically the fold change expression values derived from $\Delta\Delta$Ct calculations (Brankatschk et al., 2012). The independent variables include the genes being measured and the experimental conditions under which the samples are processed.

Microscopy data further illustrate the complexity of multidimensional datasets (Rueden et al., 2017). In this context, the dependent variable might be a quantifiable feature, such as cell count or morphological metrics extracted from image analyses. The independent variables can expand immensely to include factors such as well-plate coordinates in a 96-well plate, Z-positions in confocal microscopy, and time points in time-lapse studies.

Lastly, big-data aggregation tools like `Metascape` provide a rich source of multidimensional data by integrating various dependent variables, such as gene expression fold changes and associated p-values, with independent variables spanning gene identifiers, gene ontology terms, and

ontology classes derived from multiple databases (Y. Zhou et al., 2019). Despite the provision of summarized graphical outputs, the raw data often remain in complex, nested formats within Excel sheets, posing significant challenges for hypothesis-driven research.

This extensive integration of multiple dimensions requires sophisticated visualization and analysis techniques. While basic statistical visualizations suffice for one- or two-dimensional data, more complex data sets necessitate advanced techniques, which allow researchers to visualize and interact with data in ways that elucidate the underlying patterns and relationships (Dunn et al., 2017). However, the gap between available visualization tools and the needs of clinicians or biologists without extensive bioinformatics training remains wide, emphasizing the need for intuitive, user-friendly tools that bridge this knowledge gap and enhance the accessibility of complex data analyses (Dunn et al., 2017).

Nontransparencies in Biomedical Data Analyses

The advent of advanced technologies in biosciences has ushered in an era of *big data*, characterized by unprecedented volumes and complexities of data (Bubendorf, 2001; A. Yang et al., 2017; Ekmekci et al., 2016). This rise has been paralleled by significant challenges in data analysis, particularly impacting the reproducibility of scientific research. Studies such as the Baker (2016) survey revealed that more than 70 % of researchers have tried and failed to reproduce another scientist's experiments, highlighting a reproducibility crisis that questions the reliability of scientific findings (Begley & Ioannidis, 2015; Ioannidis, 2005).

Reproducibility is considered foundational to scientific research, ensuring that findings are reliable and verifiable. Still, its meaning requires precise definition (Goodman et al., 2016). The common understanding of scientific reproduction implies not only that detailed information is provided to enable independent repetition (*transparency*), but also that time and effort is invested into repeating the experiments (*corroboration*). However, since modern biomedical journals are demanding novelty research, and since experiments have become highly specialized and time-intensive, repeating someone else's work is considered neither interesting[3] nor possible for most publications (Flier, 2022; Peng, 2011). Hence, the meaning of reproducibility is confined to *transparency*, a concept that has been applied to many fields, including clinical trials (Goodman et al., 2016; Committee on Strategies for Responsible Sharing of Clinical Trial Data et al., 2015).

Nevertheless, there is a surprising amount of evidence for nontransparencies in biomedical data analyses: For Microarray-based miRNA profiling, raw data was not reported in more than

[3]Flier (2022): *"There are no scientists with the interest, resources, or incentives to "repeat" or confirm this vast sea of published work, so whether the findings they report are reproducible will simply never be assessed."*

40 % of 127 articles, making independent verification impossible (Witwer, 2013). The same study also found that re-analysis of data often times did not support the original conclusions. Furthermore, 44 % of 233 preclinical articles describe statistical tests insufficiently, while few don't describe them at all (Gosselin, 2021). Another study reviewed 147 papers in the field of optometrics and found that 91 % did not discuss their rationale of correcting p-values for multiple comparisons (e.g. Bonferroni correction) (Armstrong, 2014). However, given that the exact use of multiple comparisons corrections has been under debate for decades, it is reasonable to assume that researchers lack the confidence to report their technique in detail (Perneger, 1998; Moran, 2003; Sullivan & Feinn, 2021). In general, p-values are target of extreme scrutiny and also the cause of many arguments, which themselves are of questionable statistical reasoning[4] (Leek & Peng, 2015). Additionally, statistical illiteracy is a well-known problem among clinicians (Lakhlifi et al., 2023). Among biomedical researchers, 77 % state that they have not received formal training in data literacy, including visualization and public deposition of data, although they understand its high relevance (Federer et al., 2016). Correspondingly, it has been communicated that there is a lack of intuitive tools to embed computational work into publications, but also a lack of bioinformaticians to translate computation into clinics (Mesirov, 2010; Smith et al., 2018; Gómez-López et al., 2019). Therefore, nontransparencies in biomedical analyses are not only caused by a habit[5] of insufficient reporting, but could be exacerbated by the confusions caused by currently available methodologies and the lack of proper training.

Semi-Big Data: Big Enough to Cause Problems

Recent advances in big data analysis have significantly improved the standardization of both raw data availability and processing pipelines (Gomez-Cabrero et al., 2014). Particularly in RNAseq analysis, automation and the use of sophisticated software have established standards that enhance reproducibility across studies. For example, tools such as `STAR` and `HISAT` for sequence alignment, and `Cufflinks` and `DESeq` for differential expression analysis, rely on scripts that standardize processing steps to produce repeatable and verifiable results (Dobin et al., 2013; Kim et al., 2015; Trapnell et al., 2012; Love et al., 2014). These frameworks not only automate data handling but also ensure that data analysis protocols are followed consistently, reducing human error and variability between different users or laboratories.

However, this level of standardization and automation has not been mirrored in the analysis of *semi-big data*. Semi-big data, as introduced in this thesis, describes datasets that are on the cusp of manageability: substantial enough to overwhelm manual analysis methods yet not

[4] Leek & Peng (2015): *"Arguing about the P value is like focusing on a single misspelling, rather than on the faulty logic of a sentence"*

[5] Peng (2011): *"[...] old habits die hard, and many will be unwilling to discard the hours spent learning existing systems."*

sufficiently large or uniform to justify the heavy computational frameworks developed for big data. Such data are frequently generated in experiments like automated microscopy or multiplex qPCR, where the scale and complexity of the data can vary significantly depending on the experimental design and objectives (Krzywinski & Savig, 2013).

Researchers often revert to basic tools such as *Microsoft Excel* for analyzing these semi-big datasets (Incerti et al., 2019). While Excel provides familiarity and immediate accessibility, it lacks the sophisticated data handling capabilities necessary for efficient and error-free processing of complex (multidimensional) datasets. This reliance on manual methods not only makes the analysis laborious and prone to mistakes but also significantly impedes the reproducibility of research findings. The time and effort required to replicate analyses done manually mean that validating findings from semi-big data can be prohibitively challenging for peer reviewers and other researchers in the field.

Given these challenges, there is a critical need for developing new tools and frameworks specifically tailored for semi-big data. These tools should bridge the gap between the simplicity of user-friendly software like Excel and the robust, script-based automation seen in big data frameworks. By providing standardized, repeatable, and easy-to-use methods for handling complex datasets, such tools could significantly enhance the reliability and efficiency of research involving semi-big data, ultimately supporting broader scientific inquiry and verification.

The Shortcomings of Common Biomedical Analysis Tools

Interactive software systems commonly used for exploratory data analysis in biomedical research often lack mechanisms to track and reproduce the researcher's actions systematically. Even when analysis is performed using scripting languages, the integration of results from multiple packages without a coherent record of the commands and code used undermines reproducibility. This practice can obscure analysis, making it difficult, if not impossible, for other researchers to replicate the results (Leek & Peng, 2015; Peng, 2011; Mesirov, 2010; Localio et al., 2018).

A particularly illustrative example is *GraphPad Prism*, a tool ubiquitously employed across biomedical disciplines for statistical analysis. Despite its widespread use, it does contribute to data analysis nontransparencies due to *Prism*'s closed-source nature and the common journal practice of not requiring detailed methodological transparency in its usage, a practice that is common in biostatistics literature (Gosselin, 2021; Localio et al., 2018). Furthermore, *GraphPad Prism* still requires manual data entry and lacks the robustness and automation necessary for handling multidimensional or semi-big data. Although, *GraphPad Prism* is compatible *"multiple variable tables"* —similar to long-form tables known from Wickham (2014)—, but does not automatically graph these kinds of tables, but only user specified subsets (*GraphPad Prism 10*

User Guide, 2024).

Moreover, *Microsoft Excel*, another staple in data processing in biomedicine, is notoriously inadequate for handling multidimensional data and complex statistical analyses. Its limitations include poor error tracking, absence of change documentation (audit trails), and a propensity for introducing errors that often go unnoticed, such as converting gene names to dates (Ziemann et al., 2016). To compensate for these shortcomings, *Microsoft* has recently integrated a Python interpreter into *Excel*, allowing researchers to automate tasks and analyze data efficiently and correctly (Excel, 2023).

Indeed, many common tools in biomedicine allow for scripting or automation to handle semi-big data more effectively. For example, `Fiji/ImageJ`, a popular image processing platform, supports extensive macro and scripting capabilities (Rueden et al., 2017). These features enable researchers to automate batch processing of image data, streamlining tasks that would otherwise require laborious manual input. Similarly, `PyMOL`, a leading tool in protein structural biology, utilizes Python scripting to automate complex tasks, allowing for detailed molecular modeling and visualization that are reproducible and scalable across datasets (*PyMOL*, 2024; Rigsby & Parker, 2016).

Although automation scripts used in tools like `Fiji/ImageJ` and `PyMOL` improve transparency for publishing singular data analysis pipelines, they still face challenges that can impede their reproducibility (Peng, 2011; Sandve et al., 2013): These scripts sometimes require specialized software environments, where setting up dependencies and configurations can be complex enough to discourage replication efforts. Additionally, these scripts do not always provide comprehensive outputs of intermediate steps, which is crucial for verifying and understanding the progression of data analysis (Sandve et al., 2013).

On the other hand, when scripts are designed to be more generalized and distributed — for instance, as a `Fiji/ImageJ` plugin or a standalone application — they can make substantial contributions to scientific research by enabling other researchers to apply these tools to their own data sets (Narzt et al., 1998; Wilkinson et al., 2016). However, this approach also comes with its own set of challenges (Sandve et al., 2013). These generalized tools often lack comprehensive user-manuals (*documentation*) and are not thoroughly tested across different platforms or data sets, which can lead to unexpected errors that can not be fixed by the user. Moreover, even when these tools are available, they frequently suffer from low adoption rates, meaning that few people are familiar with the details of such tools, further decreasing the confidence and reproducibility in the final results.

Given these complexities, there is a pressing need for new analytical tools specifically designed for semi-big data. These tools must strike a balance between the ease of use found in basic software and the robust, analytical capabilities of more sophisticated systems. By providing

standardized workflows, comprehensive documentation, and ensuring cross-platform compatibility, these tools can significantly enhance reproducibility. They not only allow researchers to perform analyses more efficiently but also ensure that these analyses are robust, transparent, and easily verifiable by the broader scientific community.

This thesis presents a software environment developed in Python, designed to bridge this gap. It demonstrates that even minimal coding skills can be leveraged to create powerful tools that standardize and accelerate the analysis of semi-big data, ultimately fostering more reproducible and trustworthy scientific research.

Modern Standards of Software Development

A main reason to write software is to define reusable instructions for task automation (Narzt et al., 1998). The complexity of software code makes it prone to errors, which can prevent its usage by persons other than the author himself. This is a problem for the general scientific community, as the software is often essential for reproduction (Sandve et al., 2013). Hence, modern journals aim to enforce standards to software development, including software written and used by biological researchers (Smith et al., 2018). Here, we provide a brief overview of the standards utilized by `plotastic` that ensure its reliability and reproducibility by the scientific community (Peng, 2011).

Modern software development is a long-term commitment of maintaining and improving code after initial release (Boswell & Foucher, 2011). Hence, it is good practice to write the software such that it is *scalable*, *maintainable* and *usable*. *Scalability* or, to be precise, *structural scalability* means that the software can easily be expanded with new features without major modifications to its architecture (Bondi, 2000). This is achieved by writing the software in a modular fashion, where each module is responsible for a single function. *Maintainability* means that the software can easily be fixed from bugs and adapted to new requirements (Kazman et al., 2020). This is achieved by writing the code in a clear and readable manner, and by writing tests that ensure that the code works as expected (Boswell & Foucher, 2011). *Usability* is hard to define (Brooke, 1996), yet one can consider a software as usable if the commands have intuitive names and if the software's manual, termed *documentation*, is up-to-date and easy to understand for new users with minimal coding experience. A software package that has not received an update for a long time (approx. one year) could be considered abandoned. Abandoned software is unlikely to be fully functional, since it relies on other software (dependencies) that has changed in functionality or introduce bugs that were not expected by the developers of all dependencies. Together, software that's scalable, maintainable and usable requires continuous changes to its codebase. There are best practices that standardize the continuous change of the codebase, including version control, continuous integration (often referred to as CI), and software testing.

Version control is a system that records changes to the codebase line by line, documenting of the detailed history of the codebase, including the person and timepoint of every change. This is required to isolate new and experimental features into newer versions and away from the stable version that's known to work. The most popular version control system is `Git`, which is considered the industry standard for software development (Chacon & Straub, 2024). `Git` can use GitHub.com as a platform to store and host codebases in the form of software repositories. GitHub's most famous feature is called "pull request". A pull request is a request from anyone registered on GitHub to include changes to the codebase (as in *"please pull this into your main code"*). One could see pull requests as the identifying feature of the open source community, since it exposes the codebase to potentially thousands of independent developers, reaching a workforce that is impossible to achieve with closed source models used by paid software companies.

Continuous integration (CI) is a software development practice in which developers integrate code changes into a shared repository several times a day (Duvall et al., 2007). Each integration triggers the test suite, aiming to detect errors as soon as possible. The test suite includes building the software, setting up an environment for the software to run, and then executing the programmed tests, ensuring that the software runs as a whole. Continuous integration is often used together with software branches. Branches are independent copies of the codebase that are meant to be merged back into the original code once the changes are finished. Since branches accumulate multiple changes over time, this can lead to minor incompatibilities between the branches of all developers (integration conflicts), which is something that CI helps to prevent.

Continuous integration especially relies on a thorough software testing suite. Software testing is the practice of writing code that checks if the codebase works as expected (Myers et al., 2011). The main type of software testing is unit testing, which tests the smallest units of the codebase (functions and classes) in isolation (Listing 1). Software testing is automated by specialized frameworks that execute the tests and report the results, a popular example being `pytest`, which is utilized by `plotastic` (Krekel et al., 2004).

Listing 1: Example of an arbitrary Python function and its respective unit test function. The first function simply returns the number 5. The second function tests if the first function indeed returns the number 5. The test function is named with the prefix "`test_`" and is placed in a file that ends with the suffix "`_test.py`". Testing frameworks such as `pytest` scan the repository for files that end with "`_test.py`" and execute the functions that start with "`test_`". Note that code after "`#`" is considered a comment and won't be executed.

```python
# Define a function called "give_me_five" that returns the number 5
def give_me_five():
    return 5
# Define a test function asserting that "give_me_five" returns 5
def test_give_me_five():
    assert give_me_five() == 5
```

The quality of the software testing suite is measured by the code coverage, the precision of the tests, and the number of test-cases that are checked. The code coverage is the percentage of the codebase that is called by the testing functions, which should be as close to 100% as possible, although it does not measure how well the code is tested. The precision of the test is not a measurable quantity, but it represents if the tests truly checks if the code works as expected. The number of test-cases is the number of different scenarios that are checked by the testing functions, for example testing every possible option or combinations of options for functions that offer multiple options.

Together, the standards of software development, including version control, continuous integration, and software testing, ensure that the software is scalable, maintainable, and usable. This is especially important for software that is used by the scientific community, as it ensures that the software is working as expected at defined versions years after publishing scientific results.

What makes Python an "Easy" Programming Language?

Here, we provide a general overview of the Python programming language, explaining terms like *"type"*, *"method"*, etc., in order to prepare readers without prior programming experience for the following chapters. We also describe the design principles of Python to lay out the key concepts that differentiate Python compared to other programming languages. A more detailed tutorial on Python that's specialized for bioscientists is found in Ekmekci et al. 2016

Languages such as Python are considered *"high-level"*, which means that it is designed to be easy to read and write, but also independent of hardware by hiding (*"abstracting"*) underlying details (*The Python Language Reference*, 2024). A key principle of Python is the emphasis on implementing a syntax that is concise and close to human language (Listing 2, Listing 3).

Listing 2: Example of readable Python code. This one-line code returns the words (string) `"Hello, World!"` when executed. The command is straightforward and easy to understand.

```python
print("Hello, World!")
# Expected output: Hello, World!
```

Listing 3: Example of less readable code written in the low-level programming language C. This code is doing exactly the same as the Python code in Listing 2, but is harder to understand because more steps are needed, including the import of a library `stdio.h` and the definition of a function called `main`. Note that C uses `//` to begin comment sections.

```c
#include <stdio.h>          // Import functions for standard input & output
int main() {                // Define a function called 'main'
    printf("Hello, World!");
    return 0;
}
// Expected output: Hello, World!
```

Furthermore, Python is an *interpreted* language, which means that the code is executed line by line. This makes coding easier because the programmer can see the results of the code immediately after writing it, and error messages point to the exact line where the error occurred (Listing 4). This is in contrast to *compiled* languages, where the code has to be compiled into machine code before it can be executed. The advantage of compiled languages is that the code runs faster, because the machine code is optimized for the hardware.

Listing 4: Example of an error message. Since Python is case-sensitive, the error is caused by misspelling the variable name `hi` as `Hi`. The error message begins with the type of error (`NameError`), followed by a traceback that shows the sequence of function calls that led to the error. The traceback located the source of the error in the file `errormessage.py` and points to the line that caused the error. Finally, the error message explains the error, stating that `'Hi'` is undefined.

```
hi = "Hello World"  # Define a variable 'hi' with the value "Hello World"
print(Hi)           # Print the value of the undefined variable 'Hi'

# Expected Output:
# ---------------------------------------------------------------------------
# NameError                                 Traceback (most recent call last)
# File /Users/martinkuric/Documents/errormessage.py:2
#       1 hi = "Hello World"
# ----> 2 print(Hi)

# NameError: name 'Hi' is not defined
```

Python automates tasks that would otherwise require an advanced understanding of computer hardware, like the need for manual allocation of memory space. This is achieved by using a technique called *"garbage collection"*, which automatically frees memory space that is no longer needed by the program. This is a feature that is not present in low-level programming languages like C or C++, that were designed to maximize control over hardware.

Another hallmark of Python is its *dynamic typing system*. In Python the type is inferred automatically during code execution (Listing 5). This is in contrast to *statically* typed languages like C, where the type of a variable has to be declared explicitly and cannot be changed during code execution (Listing 6) (*The Python Language Reference*, 2024).

Listing 5: Example of dynamic typing in Python. The variable "`a`" is assigned the value 5, which is of type integer. The variable "`a`" is then overwritten with the value "`Hello, World!`", which is of type string. Python allows dynamic re-assignment of variables with different types. Note that code after "`#`" is considered a comment and won't be executed.

```
a = 5  # Type integer
a = 5.0  # Type float
a = 'Hello, World!'  # Type string
a = True  # Type boolean
a = False  # Type boolean
a = [1, 2, 3]  # Type list of integers
a = {'name': 'Regina'}  # Type dictionary
```

Dynamic typing makes Python a very beginner-friendly language, since one does not have to keep track of the type of each variable. However, this also makes Python a slower language,

Listing 6: Example of static typing in C. The variable "a" is declared as an integer (`int`), and can only store integers. The variable "a" is then assigned the value 5, which is an integer. The variable "a" is then assigned the value `'Hello, World!'`, which is a string. This results in a compilation error, because the variable "a" can only store integers. Note that code after "//" is considered a comment and won't be executed.

```
int a;  // Declare type as integer
a = 5;
a = 'Hello, World!';  // Compilation error!
```

because the interpreter has to check the type of each variable during code execution. Also, developing code with dynamic typing systems is prone to introducing bugs (`TypeError`), because it allows unexperienced developers to convert variables from one type to another without noticing, leading to unexpected behavior. Hence, larger Python projects require disciplined adherence to programming conventions. One such convention is *type hinting*, which is a way to explicitly note the type of a variable. Type hinting does not have an effect on the code, but it makes the code more readable and understandable for other developers, and allows for development environments to detect type errors before execution (Listing 7) (van Rossum et al., 2014).

Listing 7: Example of type hints used in Python. Explicitly stating the type of the variable is optional and does not change the behavior of the code, but behaves exactly as shown in Listing 5.

```
a: int = 5
a: str = 'Hello, World!'
```

To make Python as easy as possible, python packages aim to reduce the amount of code that has to be written by the user. For example, the package `matplotlib` is a plotting library where every command is written such that the user immediately understands its purpose, like plotting a line or labeling an axis (Listing 8). Hence `matplotlib` code is a sequence of simple function calls, where the state of the plot is modified and saved in the background line by line.

Listing 8: Example of using pre-written functions of a Python package. The functions of the package `matplotlib.pyplot` become accessable by importing the package as `plt`, where plt serves as an alias (or rather shortcut) to access the functions of the package. Then, two arbitrary lists are defined, `x` and `y`. These datapoints are plotted (scatterplot) using the function `plot` The plots x- and y-axes are then labeled and saved as an image. The code is written in a sequence of function calls, where the state of the plot is saved in the background. The plot is then displayed using the function `show`.

```
import matplotlib.pyplot as plt  # Make functions accessible via plt
x = [1, 2, 3, 4, 5]              # Define arbitrary x values
y = [1, 4, 9, 16, 25]            # Define arbitrary y values
plt.plot(x, y)                   # Plot x against y
plt.xlabel('Timepoint')          # Add a label to the x axis
plt.ylabel('Foldchange')         # Add a label to the y axis
plt.title('Gene Expression')     # Add a title above the plot
plt.savefig('plot.png')          # Save plot as image onto harddrive
plt.show()                       # Show the plot preview
```

However, when no pre-written functions or packages are available, Python offers the tools of

a general purpose programming language to write and deploy custom code easily. Programming styles can be classified into two main paradigms: *functional* and *object-oriented* programming, which can be understood as different ways to structure code. Python supports both paradigms. In *functional* programming, the code is written in a way that the program is a sequence of function calls, where each function call returns a value that is used in the next function call (Listing 9). This approach is useful when multiple actions have to be performed on the same data and the structure of the data is relatively simple, for example a string of a DNA sequence.

Listing 9: Example of functional programming in Python. The code defines a function called "`find_restriction_site`" that finds the position of a restriction site in a DNA sequence. The function "`cut`" uses the function "`find_restriction_site`" to cut the sequence at the restriction site. To execute both functions, we first define an arbitrary DNA sequence and then call the function `cut` passing the sequence as an argument.

```
def find_restriction_site(sequence: str):  # Define a function
    return sequence.find('GCGC')           # Find the position of 'GCGC'

def cut(sequence: str):                        # Define another function
    position = find_restriction_site(sequence) # Use the function above
    return sequence[position:]                 # Cut sequence at the position

seq1 = 'TGAGCTGAGCTGATGCGCTATATTTAGGCG'  # Define an arbitrary sequence
seq1_cut = cut(seq1)                     # Cut the sequence
print(seq1_cut)                          # Show the result
# Expected output: GCGCTATATTTAGGCG
```

When the data itself gains in complexity, for example when storing not just the gene sequence, but also the promotor sequence, an *object-oriented* approach is more suitable (Listing 10). Object-oriented programming is a programming paradigm that uses objects and classes. An object is a collection of both data and functions, and a class is a blueprint for creating objects. The data of an object is stored as *attributes*. Functions that are associated with an object are called *methods*. A major benefit of using an object oriented versus a functional approach is that the data itself programmable, enabling the programmer to define the behavior of the data through methods. This is achieved by using '`self`' to reference the objects themselves inside the class. `self` can be understood as *"this object"*, and is a placeholder for objects that are to be created from the blueprint. `self` is required to access attributes and methods before specific objects are created in order to program how the objects are to be changed when calling methods.

Listing 10: Example of object oriented programming in Python. The class is called "`Gene`" and acts as a blueprint to create `Gene` objects. `Gene` has four methods, "`__init__`", "`find_promotor`", "`find_restriction_site`" and "`cut`". The method "`__init__`" is called when creating ("initializing") an object, which fills the object with user-defined data. The parameter "`self`" is a placeholder for the objects that are to be created. "`find_promotor`" is a method that finds the position of the promotor in the gene and is called during object initialization.

```
class Gene:                                # Define a Gene class
    def __init__(self, sequence: str):     # Define how a Gene object is created
```

```
        self.sequence: str = sequence     # Save sequence as attribute
        self.promotor: str = self.find_promotor() # Automatically find promotor
        # Add further Gene attributes here
    def find_promotor(self):              # Define how to find the promotor
        return self.sequence.find('TATA')
    def find_restriction_site(self):      # Define how to find restriction site
        return self.sequence.find('GCGC') # Find the position of 'GCGC'
    def cut(self):                        # Define how to cut the gene
        position = self.find_restriction_site() # Call the method above
        return self.sequence[position:]   # Cut the gene at the position

gene1 = Gene(sequence='TGAGCTGAGCTGATGCGCTATATTTAGGCG') # Create Gene object
gene1_cut = gene1.cut()                   # Call the method cut
print(gene1_cut)                          # Show result
# Expected output: GCGCTATATTTAGGCG
```

When designing software, both functional and object oriented programming can be used together, where object oriented programming is often used to design the program's overall architecture, and functional programming is used to implement the algorithms of the program's features. This allows for scalability of the software, as every single class is extended through the addition of new methods. Furthermore, classes can be expanded in their functionalities through *inheritance* (Listing 11). Inheritance is a feature of object-oriented programming that allows a class to access every attribute and method of a parent class. For example, one could extend the class "Gene" with a class "mRNA", by writing a class "mRNA" that inherits from the class "Gene".

Listing 11: Example of inheritance in Python. The class "mRNA" inherits from the class "Gene". The class "mRNA" has two methods, "__init__" and "find_stopcodon". The method "find_stopcodon" finds the position of stop codons.

```
class mRNA(Gene):          # Define the mRNA class, inheriting from Gene class
    def __init__(self, sequence: str):   # Define how an mRNA object is created
        super().__init__(sequence)       # Get attributes from parent class
        self.sequence.replace('T', 'U')  # Replace thymine with uracil
    def find_stopcodons(self):           # Define how to find stop codons
        return self.sequence.find('UGA') # Find the position of 'UGA'

mrna1 = mRNA(sequence='TGAGCTGAGCTGATGCGCTATATTTAGGCG') # Create mRNA object
print(mrna1.find_stopcodons())              # Show the position of stop codons
# Expected output: [0, 5, 10]
```

Together, Python is not just beginner-friendly, but also well respected for its ease in development, which is why it is widely used in professional settings for web development, data analysis, machine learning, biosciences and more (Ekmekci et al., 2016; Rayhan & Gross, 2023).

The Potential of Python Data Science Packages for Biomedicine

Python includes a vast number of built-in packages used for basic data-types, software development, simple math operations, etc., (*The Python Language Reference*, 2024). Still, Python relies on packages developed by its users to provide specialized tools for data analysis. A Python

package consists of multiple Python *modules*, where each module is a text-file with a `.py` ending containing Python code. Famous examples of such packages are `pytorch` and `tensorflow`, that are used to build models of artificial intelligence, including *ChatGPT* (Paszke et al., 2019; Abadi et al., 2016; Radford et al., 2019). Here, we outlay the most important packages used for `plotastic` in Chapter 2 and present examples how these packages are utilized in modern biomedical research.

`Interactive Python`: The standard Python interface is insufficient for data science, because it lacks the tools to quickly and conveniently visualize and explore data. `IPython` can be understood as an enhanced version of the standard Python interpreter, designed to improve the interactivity of Python code execution (Perez & Granger, 2007). `IPython` introduces features like rich media support to display graphics, but also helps users to use correct python data types through dynamic type introspection, detecting errors in the code. This functionality is akin to what *MATLAB* and *RStudio* provide through their advanced graphical user interfaces and extensive debugging tools. `IPython` is most often utilized in the form of *Jupyter Notebooks*.

`Jupyter`: `Jupyter` is an evolution of `IPython`, introducing the *Jupyter notebook* format, which has the file-ending `.ipynb` (Kluyver et al., 2016). Jupyter Notebooks are documents that combine both code and text structured as *code cells* and *markdown cells*, respectively. Markdown cells allow the author to provide additional information with text formatting, for example structuring the document with headings and subheadings, adding hyperlinks, images and mathematical formulas. Code cells can be executed individually, displaying the output directly below the cell. This allows for an interactive exploration of data, but also makes Jupyter Notebooks a very human-readable format that outlays data analysis in a clear manner with precise and reproducible documentation of all data processing steps. Another major benefit of Jupyter Notebooks are interchangable *Kernels*, allowing the execution of code in different programming languages, such as R, Julia, and C++ (Giorgi et al., 2022). Today, Jupyter Notebooks have become a standard format compatible with collaborative platforms like *Google Colab* and *JupyterLab*, but also professional software development tools like *VS Code*, and *PyCharm*. For biomedical research, Jupyter Notebooks are a powerful solution for improving reproducibility: They elegantly combine both documentation and code execution into a concise presentation of the data analysis process, hence being an intuitive tool to both capture and embed computational work directly into papers, a requirement postulated by Mesirov (2010). Jupyter notebooks are increasingly found in the supplemental of modern publications of both bioinformatics and wet-lab research (Taskiran et al., 2024; Bosch-Queralt et al., 2022; Howe & Chain, 2015).

`NumPy`: Central processing units (CPU) usually execute one instruction on one data point at a time. For manipulating tabular data, this is inefficient as the same instruction must be repeat-

edly loaded for every data point. NumPy accelerates the mathematical capabilities of Python by enabling large-scale operations on multi-dimensional arrays and matrices with high efficiency (Harris et al., 2020). One key feature of NumPy is the implementation of "vectorization" or SIMD (Single Instruction, Multiple Data) instructions. SIMD allows multiple data points to be processed simultaneously, significantly speeding up operations that are inherently parallelizable, such as matrix addition or multiplication. NumPy's syntax and functional approach to array manipulation have set a standard for matrix computation, influencing the design of advanced AI frameworks such as PyTorch and mlx, which mirrors several of NumPy's functionalities to facilitate ease of use for those familiar with NumPy (Paszke et al., 2019; Hannun et al., 2023). This standardization has made NumPy an attractive tool not only in genomics (Ding et al., 2023), but also for modern clinical applications like imaging technologies and augmented-reality in surgery (Thompson et al., 2020).

Pandas: Tables are the most common way to store experimental results. Pandas extends Python with a tabular datatype, called DataFrame, which allows for easy data manipulation with integrated indexing (Mckinney, 2011). The intuitive interface of Pandas can be likened to *Microsoft Excel*; however, it is vastly more powerful due to its speed, functionality, and ability to handle larger datasets, e.g. by running efficient numpy vectorization in the background. Unlike *Excel*, Pandas enables automation by summarizing processing commands into scripts, documenting each step, and ensuring reproducibility. Pandas is used in biomedicine for data wrangling, data cleaning, and data analysis, as it allows for the integration of multiple data sources into a single table (Santos et al., 2020).

matplotlib: matplotlib is a plotting library that provides a wide range of static, animated, and interactive plots and graphs Listing 8 (Hunter, 2007). It serves as the foundation for many visualization tools and is particularly valued for its flexibility and customization options. For example, Pandas uses matplotlib to plot column datapoints directly from a DataFrame object, creating histograms or scatter plots, which is useful for preliminary data analysis and checking data distributions. However, matplotlib uses a low-level syntax, hence plots generated by matplotlib can be cumbersome to format and customize.

seaborn: While the low-level syntax of matplotlib is valued for its flexibility, formatting publication grade plots can be laborious, and its inconsistent syntax can make it difficult to remember the correct commands for different plot types. seaborn is a high-level interface on top of matplotlib that offers a more intuitive and highly standardized syntax across a wide array of plot types (Waskom, 2021). seaborn also integrates closely with Pandas data structures: It automatically groups datapoints, calculates measures of both central tendency (e.g. mean, median) and variance (e.g. standard deviation), and displays them into the plot (e.g. error bars). This completely replaces manual calculation of descriptive statistics. seaborn also offers

intuitive grouping (*facetting*) of data points, which simplifies the creation of complex visualizations involving multidimensional data, making it easier to reveal patterns and relationships via color encoding, faceting, and automated statistical fits. This is particularly useful in biomedical research for visualizing and understanding complex datasets, such as large quantities of protein data (Krzywinski & Savig, 2013; Weiss, 2022). `seaborn` could indirectly contribute to improving reproducibility in biomedical research by making visualizations of complex data very accessible through an easy and standardized syntax.

`Pingouin`: Integrating both data visualization and statistical analysis is beneficial for researchers who wish to conduct advanced statistical analysis without switching between different software environments. `Pingouin` is designed to be a user-friendly statistical tool that offers a straightforward syntax for performing statistical tests, which are commonly implemented in R (Vallat, 2018). Unlike R, `Pingouin` integrates seamlessly within the Python ecosystem, which allows combining data manipulation, analysis, and visualization all in one platform. This improves reproducibility by reducing the number of software tools required to analyze data. Despite its potential to streamline the data analysis process, `Pingouin` has not been widely adopted by biomedical research, yet. One example of a study that utilized `Pingouin` is the work of Kelly et al. (2023) in the field of Patient Public Involvement (PPI), producing an ethical matrix that allows for the inclusion of stakeholder opinion in medical research design. This lack of `Pingouin`'s adoption in biomedicine could be due to recent development and the dominance of R in the field. However, since Python offers multiple benefits over R in syntax, software development, runtime performance and integration with other tools (like including performant C++ code), `Pingouin` is an attractive standard for future statistical analyses in biomedicine (Gorelick & Ozsvald, 2020).

Together, these python packages form the backbone of modern data analysis in Python, often times combining software from different languages to accelerate certain features, while retaining the ease of use and readability that Python is known for. This is particularly advantageous in the field of biomedicine, where the requirements of modern data analysis are often complex and require a high degree of flexibility and customization.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., … Zheng, X. (2016, March). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems* (No. arXiv:1603.04467). arXiv. Retrieved 2024-03-07, from http://arxiv.org/abs/1603.04467 doi: 10.48550/arXiv.1603.04467

Abdallah, N. H., Lakshman, A., Kumar, S. K., Cook, J., Binder, M., Kapoor, P., … Rajkumar, S. V. (2024, January). Mode of progression in smoldering multiple myeloma: A study of 406 patients. *Blood Cancer Journal*, *14*(1), 1–7. Retrieved 2024-05-22, from https://www.nature.com/articles/s41408-024-00980-5 doi: 10.1038/s41408-024-00980-5

Aggarwal, R., Ghobrial, I. M., & Roodman, G. D. (2006, October). Chemokines in multiple myeloma. *Experimental hematology*, *34*(10), 1289–1295. Retrieved 2023-04-02, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3134145/ doi: 10.1016/j.exphem.2006.06.017

Akhmetzyanova, I., McCarron, M. J., Parekh, S., Chesi, M., Bergsagel, P. L., & Fooksman, D. R. (2020). Dynamic CD138 surface expression regulates switch between myeloma growth and dissemination. *Leukemia*, *34*(1), 245–256. Retrieved 2023-04-04, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6923614/ doi: 10.1038/s41375-019-0519-4

Alcorta-Sevillano, N., Macías, I., Infante, A., & Rodríguez, C. I. (2020, December). Deciphering the Relevance of Bone ECM Signaling. *Cells*, *9*(12), 2630. Retrieved 2023-12-20, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7762413/ doi: 10.3390/cells9122630

Alsayed, Y., Ngo, H., Runnels, J., Leleu, X., Singha, U. K., Pitsillides, C. M., … Ghobrial, I. M. (2007, April). Mechanisms of regulation of CXCR4/SDF-1 (CXCL12)-dependent migration and homing in multiple myeloma. *Blood*, *109*(7), 2708–2717. doi: 10.1182/blood-2006-07-035857

Anders, S., Pyl, P. T., & Huber, W. (2015, January). HTSeq–a Python framework to work with high-throughput sequencing data. *Bioinformatics (Oxford, England)*, *31*(2), 166–169. doi: 10.1093/bioinformatics/btu638

Andrews, S. (2010). *FASTQC. A quality control tool for high throughput sequence data.*

Arefin, S. E., Heya, T. A., Al-Qudah, H., Ineza, Y., & Serwadda, A. (2023, July). *Unmasking the giant: A comprehensive evaluation of ChatGPT's proficiency in coding algorithms and data structures* (No. arXiv:2307.05360). arXiv. Retrieved 2024-05-03, from http://arxiv.org/abs/2307.05360 doi: 10.48550/arXiv.2307.05360

Armstrong, R. A. (2014, September). When to use the Bonferroni correction. *Ophthalmic & Physiological Optics: The Journal of the British College of Ophthalmic Opticians (Optometrists)*, *34*(5), 502–508. doi: 10.1111/opo.12131

Asosingh, K., Günthert, U., De Raeve, H., Van Riet, I., Van Camp, B., & Vanderkerken, K. (2001). A unique pathway in the homing of murine multiple myeloma cells: CD44v10 mediates binding to bone marrow endothelium. *Cancer Research*, *61*(7), 2862–2865.

Baker, M. (2016, May). 1,500 scientists lift the lid on reproducibility. *Nature*, *533*(7604), 452–454. Retrieved 2024-04-22, from https://www.nature.com/articles/533452a doi: 10.1038/533452a

Bao, L., Lai, Y., Liu, Y., Qin, Y., Zhao, X., Lu, X., … Huang, X. (2013, September). CXCR4 is a good survival prognostic indicator in multiple myeloma patients. *Leukemia Research*, *37*(9), 1083–1088. doi: 10.1016/j.leukres.2013.06.002

Barzilay, R., Ben-Zur, T., Bulvik, S., Melamed, E., & Offen, D. (2009, May). Lentiviral delivery of LMX1a enhances dopaminergic phenotype in differentiated human bone marrow mesenchymal stem cells. *Stem cells and development*, *18*(4), 591–601. doi: 10.1089/scd.2008.0138

Begley, C. G., & Ioannidis, J. P. A. (2015, January). Reproducibility in science: Improving the standard for basic and preclinical research. *Circulation Research*, *116*(1), 116–126. doi: 10.1161/CIRCRESAHA.114.303819

Bianco, P. (2014). "Mesenchymal" stem cells. *Annual review of cell and developmental biology*, *30*, 677–704. doi: 10.1146/annurev-cellbio-100913-013132

Bladé, J., Beksac, M., Caers, J., Jurczyszyn, A., von Lilienfeld-Toal, M., Moreau, P., … Richardson, P. (2022, March). Extramedullary disease in multiple myeloma: A systematic literature review. *Blood Cancer Journal*, *12*(3), 1–10. Retrieved 2023-03-24, from https://www.nature.com/articles/s41408-022-00643-3 doi: 10.1038/s41408-022-00643-3

Blonska, M., Zhu, Y., Chuang, H. H., You, M. J., Kunkalla, K., Vega, F., & Lin, X. (2015, February). Jun-regulated genes promote interaction of diffuse large B-cell lymphoma with the microenvironment. *Blood*, *125*(6), 981–991. Retrieved 2023-03-01, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4319238/ doi: 10.1182/blood-2014-04-568188

Bobin, A., & Leleu, X. (2022, September). Recent advances in the treatment of multiple myeloma: A brief review. *Faculty Reviews*, *11*, 28. Retrieved 2024-03-27, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9523543/ doi: 10.12703/r/11-28

Bolado-Carrancio, A., Rukhlenko, O. S., Nikonova, E., Tsyganov, M. A., Wheeler, A., Garcia-Munoz, A., … Kholodenko, B. N. (2020, July). Periodic propagating waves coordinate RhoGTPase network dynamics at the leading and trailing edges during cell migration. *eLife*, *9*, e58165. Retrieved 2024-04-25, from https://elifesciences.org/articles/58165 doi: 10.7554/eLife.58165

Bondi, A. B. (2000, September). Characteristics of scalability and their impact on performance. In *Proceedings of the 2nd international workshop on Software and performance* (pp. 195–203). New York, NY, USA: Association for Computing Machinery. Retrieved 2024-03-07, from https://dl.acm.org/doi/10.1145/350391.350432 doi: 10.1145/350391.350432

Bosch-Queralt, M., Tiwari, V., Damkou, A., Vaculčiaková, L., Alexopoulos, I., & Simons, M. (2022, March). A fluorescence microscopy-based protocol for volumetric measurement of lysolecithin lesion-associated de- and re-myelination in mouse brain. *STAR protocols*, *3*(1), 101141. doi: 10.1016/j.xpro.2022.101141

Boswell, D., & Foucher, T. (2011). *The Art of Readable Code: Simple and Practical Techniques for Writing Better Code*. "O'Reilly Media, Inc.".

Bou Zerdan, M., Nasr, L., Kassab, J., Saba, L., Ghossein, M., Yaghi, M., … Chaulagain, C. P. (2022). Adhesion molecules in multiple myeloma oncogenesis and targeted therapy. *International Journal of Hematologic Oncology*, *11*(2), IJH39. Retrieved 2023-02-01, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9136637/ doi: 10.2217/ijh-2021-0017

Brandl, A., Solimando, A. G., Mokhtari, Z., Tabares, P., Medler, J., Manz, H., … Beilhack, A. (2022, March). Junctional adhesion molecule C expression specifies a CD138low/neg multiple myeloma cell population in mice and humans. *Blood Advances*, *6*(7), 2195–2206. Retrieved 2023-04-04, from https://doi.org/10.1182/bloodadvances.2021004354 doi: 10.1182/bloodadvances.2021004354

Brankatschk, R., Bodenhausen, N., Zeyer, J., & Bürgmann, H. (2012, June). Simple Absolute Quantification Method Correcting for Quantitative PCR Efficiency Variations for Microbial Community Samples. *Applied and Environmental Microbiology*, *78*(12), 4481–4489. Retrieved 2023-05-27, from https://journals.asm.org/doi/10.1128/AEM.07878-11 doi: 10.1128/AEM.07878-11

Brooke, J. (1996, January). SUS – a quick and dirty usability scale. In (pp. 189–194).

Bubendorf, L. (2001, August). High-throughput microarray technologies: From genomics to clinics. *European Urology*, *40*(2), 231–238. doi: 10.1159/000049777

Burger, R., Guenther, A., Bakker, F., Schmalzing, M., Bernand, S., Baum, W., … Gramatzki, M. (2001). Gp130 and ras mediated signaling in human plasma cell line INA-6: A cytokine-regulated tumor model for

plasmacytoma. *The Hematology Journal: The Official Journal of the European Haematology Association*, *2*(1), 42–53. doi: 10.1038/sj.thj.6200075

Burger, R., Günther, A., Bakker, F., Schmalzing, M., Bernand, S., Baum, W., … Gramatzki, M. (2001, January). Gp130 and ras mediated signaling in human plasma cell line INA6: A cytokine-regulated tumor model for plasmacytoma. *Hematology Journal - HEMATOL J*, *2*, 42–53. doi: 10.1038/sj.thj.6200075

Bustin, S. A. (2014, December). The reproducibility of biomedical research: Sleepers awake! *Biomolecular Detection and Quantification*, *2*, 35–42. Retrieved 2024-03-18, from https://www.sciencedirect.com/science/article/pii/S2214753515000030 doi: 10.1016/j.bdq.2015.01.002

Bustin, S. A., Benes, V., Garson, J., Hellemans, J., Huggett, J., Kubista, M., … Vandesompele, J. (2013, November). The need for transparency and good practices in the qPCR literature. *Nature Methods*, *10*(11), 1063–1067. Retrieved 2024-05-16, from https://www.nature.com/articles/nmeth.2697 doi: 10.1038/nmeth.2697

Caplan, A. (1991). Mesenchymal stem cells. *Journal of orthopaedic research : official publication of the Orthopaedic Research Society*, *9*(5), 641–50. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/1870029 doi: 10.1002/jor.1100090504

Caplan, A. I. (1994, July). The mesengenic process. *Clinics in plastic surgery*, *21*(3), 429–435.

Carlson, M. (2016). Org.Hs.eg.db. *Bioconductor*. Retrieved 2023-06-09, from http://bioconductor.org/packages/org.Hs.eg.db/ doi: 10.18129/B9.bioc.org.Hs.eg.db

Chacon, S., & Straub, B. (2024, March). *Git - Book*. Retrieved 2024-03-07, from https://git-scm.com/book/de/v2

Charlier, F., Weber, M., Izak, D., Harkin, E., Magnus, M., Lalli, J., … Repplinger, S. (2022, October). *Trevis-md/statannotations: V0.5*. Zenodo. Retrieved 2023-11-16, from https://zenodo.org/record/7213391 doi: 10.5281/ZENODO.7213391

Chatterjee, M., Hönemann, D., Lentzsch, S., Bommert, K., Sers, C., Herrmann, P., … Bargou, R. C. (2002, November). In the presence of bone marrow stromal cells human multiple myeloma cells become independent of the IL-6/gp130/STAT3 pathway. *Blood*, *100*(9), 3311–3318. doi: 10.1182/blood-2002-01-0102

Codecov. (2024). Retrieved 2024-05-02, from https://github.com/codecov

Committee on Strategies for Responsible Sharing of Clinical Trial Data, Board on Health Sciences Policy, & Institute of Medicine. (2015). *Sharing Clinical Trial Data: Maximizing Benefits, Minimizing Risk*. Washington (DC): National Academies Press (US). Retrieved 2024-04-23, from http://www.ncbi.nlm.nih.gov/books/NBK269030/

Cooper, G. M. (2000). The Cell: A Molecular Approach. 2nd Edition. *Sinauer Associates*, Proliferation in Development and Differentiation. Retrieved from https://www.ncbi.nlm.nih.gov/books/NBK9906/

da Silva Meirelles, L., Chagastelles, P. C., & Nardi, N. B. (2006, June). Mesenchymal stem cells reside in virtually all post-natal organs and tissues. *Journal of cell science*, *119*(Pt 11), 2204–2213. doi: 10.1242/jcs.02932

Davidson-Pilon, C. (2019, August). Lifelines: Survival analysis in Python. *Journal of Open Source Software*, *4*(40), 1317. Retrieved 2024-05-02, from https://joss.theoj.org/papers/10.21105/joss.01317 doi: 10.21105/joss.01317

Depil, S., Duchateau, P., Grupp, S. A., Mufti, G., & Poirot, L. (2020, March). 'Off-the-shelf' allogeneic CAR T cells: Development and challenges. *Nature Reviews Drug Discovery*, *19*(3), 185–199. Retrieved 2024-03-27, from https://www.nature.com/articles/s41573-019-0051-2 doi: 10.1038/s41573-019-0051-2

Ding, W., Goldberg, D., & Zhou, W. (2023, August). PyComplexHeatmap: A Python package to visualize multimodal genomics data. *iMeta*, *2*(3), e115. doi: 10.1002/imt2.115

Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., … Gingeras, T. R. (2013, January). STAR: Ultrafast universal RNA-seq aligner. *Bioinformatics*, *29*(1), 15–21. Retrieved 2023-05-27, from

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3530905/ doi: 10.1093/bioinformatics/bts635

Doddi, S., & Rashid, M. H. (2024). Disparities in Multiple Myeloma Mortality Rate Trends by Demographic Status in the USA. *Cancer Diagnosis & Prognosis*, *4*(3), 288–294. doi: 10.21873/cdp.10322

Dominici, M., Le Blanc, K., Mueller, I., Slaper-Cortenbach, I., Marini, F., Krause, D., … Horwitz, E. (2006). Minimal criteria for defining multipotent mesenchymal stromal cells. The International Society for Cellular Therapy position statement. *Cytotherapy*, *8*(4), 315–317. doi: 10.1080/14653240600855905

Dotterweich, J., Schlegelmilch, K., Keller, A., Geyer, B., Schneider, D., Zeck, S., … Schütze, N. (2016, December). Contact of myeloma cells induces a characteristic transcriptome signature in skeletal precursor cells -Implications for myeloma bone disease. *Bone*, *93*, 155–166. doi: 10.1016/j.bone.2016.08.006

Dunn, W., Burgun, A., Krebs, M.-O., & Rance, B. (2017, November). Exploring and visualizing multidimensional data in translational research platforms. *Briefings in Bioinformatics*, *18*(6), 1044–1056. Retrieved 2024-04-23, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5862238/ doi: 10.1093/bib/bbw080

Duvall, P., Matyas, S., & Glover, A. (2007). *Continuous integration: Improving software quality and reducing risk*. Pearson Education. Retrieved from https://books.google.de/books?id=PV9qfEdv9L0C

Dziadowicz, S. A., Wang, L., Akhter, H., Aesoph, D., Sharma, T., Adjeroh, D. A., … Hu, G. (2022, January). Bone Marrow Stroma-Induced Transcriptome and Regulome Signatures of Multiple Myeloma. *Cancers*, *14*(4), 927. Retrieved 2022-10-25, from https://www.mdpi.com/2072-6694/14/4/927 doi: 10.3390/cancers14040927

Ekmekci, B., McAnany, C. E., & Mura, C. (2016, July). An Introduction to Programming for Bioscientists: A Python-Based Primer. *PLOS Computational Biology*, *12*(6), e1004867. Retrieved 2024-03-10, from https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004867 doi: 10.1371/journal.pcbi.1004867

Engelhardt, M., Kortüm, K. M., Goldschmidt, H., & Merz, M. (2024, February). Functional cure and long-term survival in multiple myeloma: How to challenge the previously impossible. *Haematologica*. doi: 10.3324/haematol.2023.283058

Evers, M., Schreder, M., Stühmer, T., Jundt, F., Ebert, R., Hartmann, T. N., … Leich, E. (2023, March). Prognostic value of extracellular matrix gene mutations and expression in multiple myeloma. *Blood Cancer Journal*, *13*(1), 43. doi: 10.1038/s41408-023-00817-7

Ewels, P., Magnusson, M., Lundin, S., & Käller, M. (2016, October). MultiQC: Summarize analysis results for multiple tools and samples in a single report. *Bioinformatics*, *32*(19), 3047–3048. Retrieved 2023-06-09, from https://doi.org/10.1093/bioinformatics/btw354 doi: 10.1093/bioinformatics/btw354

Excel, M. (2023, August). *Announcing Python in Excel: Combining the power of Python and the flexibility of Excel*. Retrieved 2024-03-11, from https://techcommunity.microsoft.com/t5/excel-blog/announcing-python-in-excel-combining-the-power-of-python-and-the/ba-p/3893439

Fazeli, P. K., Horowitz, M. C., MacDougald, O. A., Scheller, E. L., Rodeheffer, M. S., Rosen, C. J., & Klibanski, A. (2013, March). Marrow Fat and Bone—New Perspectives. *The Journal of Clinical Endocrinology and Metabolism*, *98*(3), 935–945. Retrieved 2023-12-20, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3590487/ doi: 10.1210/jc.2012-3634

Federer, L. M., Lu, Y.-L., & Joubert, D. J. (2016, January). Data literacy training needs of biomedical researchers. *Journal of the Medical Library Association : JMLA*, *104*(1), 52–57. Retrieved 2024-04-24, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4722643/ doi: 10.3163/1536-5050.104.1.008

Fermand, J.-P., Bridoux, F., Dispenzieri, A., Jaccard, A., Kyle, R. A., Leung, N., & Merlini, G. (2018, October). Monoclonal gammopathy of clinical significance: A novel concept with therapeutic implications. *Blood*, *132*(14), 1478–1485. doi: 10.1182/blood-2018-04-839480

Fernandez-Rebollo, E., Mentrup, B., Ebert, R., Franzen, J., Abagnale, G., Sieben, T., … Wagner, W. (2017, July). Human Platelet Lysate versus Fetal Calf Serum: These Supplements Do Not Select for Different

Mesenchymal Stromal Cells. *Scientific Reports*, *7*, 5132. Retrieved 2023-05-02, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5506010/ doi: 10.1038/s41598-017-05207-1

Flier, J. S. (2022). The Problem of Irreproducible Bioscience Research. *Perspectives in Biology and Medicine*, *65*(3), 373–395. doi: 10.1353/pbm.2022.0032

Forster, S., & Radpour, R. (2022, July). Molecular Impact of the Tumor Microenvironment on Multiple Myeloma Dissemination and Extramedullary Disease. *Frontiers in Oncology*, *12*. Retrieved 2024-05-23, from https://www.frontiersin.org/journals/oncology/articles/10.3389/fonc.2022.941437/full doi: 10.3389/fonc.2022.941437

Frassanito, M. A., Cusmai, A., Iodice, G., & Dammacco, F. (2001, January). Autocrine interleukin-6 production and highly malignant multiple myeloma: Relation with resistance to drug-induced apoptosis. *Blood*, *97*(2), 483–489. doi: 10.1182/blood.v97.2.483

Friedenstein, A., & Kuralesova, A. I. (1971, August). Osteogenic precursor cells of bone marrow in radiation chimeras. *Transplantation*, *12*(2), 99–108.

Friedenstein, A. J., Piatetzky-Shapiro, I. I., & Petrakova, K. V. (1966, December). Osteogenesis in transplants of bone marrow cells. *Journal of embryology and experimental morphology*, *16*(3), 381–390.

Gabr, M. M., Zakaria, M. M., Refaie, A. F., Ismail, A. M., Abou-El-Mahasen, M. A., Ashamallah, S. A., … Ghoneim, M. A. (2013). Insulin-producing cells from adult human bone marrow mesenchymal stem cells control streptozotocin-induced diabetes in nude mice. *Cell transplantation*, *22*(1), 133–145. doi: 10.3727/096368912X647162

Gao, D., Ji, L., Bai, Z., Ouyang, M., Li, P., Mao, D., … Shou, M. Z. (2024, January). *ASSISTGUI: Task-Oriented Desktop Graphical User Interface Automation* (No. arXiv:2312.13108). arXiv. Retrieved 2024-05-16, from http://arxiv.org/abs/2312.13108 doi: 10.48550/arXiv.2312.13108

Gao, S., Wang, Y.-T., Ma, G.-Y., Lu, M.-Q., Chu, B., Shi, L., … Bao, L. (2024, April). Solitary bone plasmacytoma: Long-term clinical outcomes in a single center. *Current Problems in Cancer*, *50*, 101095. doi: 10.1016/j.currproblcancer.2024.101095

Garcés, J.-J., Simicek, M., Vicari, M., Brozova, L., Burgos, L., Bezdekova, R., … Paiva, B. (2020, February). Transcriptional profiling of circulating tumor cells in multiple myeloma: A new model to understand disease dissemination. *Leukemia*, *34*(2), 589–603. doi: 10.1038/s41375-019-0588-4

García-Ortiz, A., Rodríguez-García, Y., Encinas, J., Maroto-Martín, E., Castellano, E., Teixidó, J., & Martínez-López, J. (2021, January). The Role of Tumor Microenvironment in Multiple Myeloma Development and Progression. *Cancers*, *13*(2). Retrieved 2021-02-02, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7827690/ doi: 10.3390/cancers13020217

Gentleman. (n.d.). *Bioconductor - BiocViews.* Retrieved 2023-06-09, from https://bioconductor.org/packages/3.17/BiocViews.html

Ghobrial, I. M. (2012, July). Myeloma as a model for the process of metastasis: Implications for therapy. *Blood*, *120*(1), 20–30. Retrieved 2022-10-15, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3390959/ doi: 10.1182/blood-2012-01-379024

Giorgi, F. M., Ceraolo, C., & Mercatelli, D. (2022, April). The R Language: An Engine for Bioinformatics and Data Science. *Life*, *12*(5), 648. Retrieved 2024-04-21, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9148156/ doi: 10.3390/life12050648

Glavey, S. V., Naba, A., Manier, S., Clauser, K., Tahri, S., Park, J., … Ghobrial, I. M. (2017, November). Proteomic characterization of human multiple myeloma bone marrow extracellular matrix. *Leukemia*, *31*(11), 2426–2434. Retrieved 2023-09-05, from https://www.nature.com/articles/leu2017102 doi: 10.1038/leu.2017.102

Gomez-Cabrero, D., Abugessaisa, I., Maier, D., Teschendorff, A., Merkenschlager, M., Gisel, A., … Tegnér,

J. (2014, March). Data integration in the era of omics: Current and future challenges. *BMC Systems Biology*, *8*(2), I1. Retrieved 2024-03-18, from https://doi.org/10.1186/1752-0509-8-S2-I1 doi: 10.1186/1752-0509-8-S2-I1

Gómez-López, G., Dopazo, J., Cigudosa, J. C., Valencia, A., & Al-Shahrour, F. (2019, May). Precision medicine needs pioneering clinical bioinformaticians. *Briefings in Bioinformatics*, *20*(3), 752–766. doi: 10.1093/bib/bbx144

Goodman, S. N., Fanelli, D., & Ioannidis, J. P. A. (2016, June). What does research reproducibility mean? *Science Translational Medicine*, *8*(341), 341ps12-341ps12. Retrieved 2024-03-18, from https://www.science.org/doi/10.1126/scitranslmed.aaf5027 doi: 10.1126/scitranslmed.aaf5027

Gorelick, M., & Ozsvald, I. (2020). *High Performance Python: Practical Performant Programming for Humans*. "O'Reilly Media, Inc.".

Gosselin, R.-D. (2021, February). Insufficient transparency of statistical reporting in preclinical research: A scoping review. *Scientific Reports*, *11*(1), 3335. Retrieved 2024-03-11, from https://www.nature.com/articles/s41598-021-83006-5 doi: 10.1038/s41598-021-83006-5

Gramatzki, M., Burger, R., Trautman, U., Marschalek, R., Lorenz, H., Hansen-Hagge, T., … Kalden, J. (1994). Two new interleukin-6 dependent plasma cell lines carrying a chromosomal abnormality involving the IL-6 gene locus. , *84 Suppl. 1*, 173a-173a. Retrieved 2023-03-24, from https://www.cellosaurus.org/cellopub/CLPUB00060

GraphPad Prism 10 User Guide. (2024). Retrieved 2024-05-14, from https://www.graphpad.com/guides/prism/latest/user-guide/multiple-variable-tables.htm

Greenstein, S., Krett, N. L., Kurosawa, Y., Ma, C., Chauhan, D., Hideshima, T., … Rosen, S. T. (2003, April). Characterization of the MM.1 human multiple myeloma (MM) cell lines: A model system to elucidate the characteristics, behavior, and signaling of steroid-sensitive and -resistant MM cells. *Experimental Hematology*, *31*(4), 271–282. doi: 10.1016/s0301-472x(03)00023-7

Gronthos, S., Graves, S. E., Ohta, S., & Simmons, P. J. (1994, December). The STRO-1+ fraction of adult human bone marrow contains the osteogenic precursors. *Blood*, *84*(12), 4164–4173.

Hannun, A., Digani, J., Katharopoulos, A., & Collobert, R. (2023). *MLX: Efficient and flexible machine learning on Apple silicon.* Retrieved from https://github.com/ml-explore

Harrington, D. P., & Fleming, T. R. (1982). A Class of Rank Test Procedures for Censored Survival Data. *Biometrika*, *69*(3), 553–566. Retrieved 2023-08-07, from https://www.jstor.org/stable/2335991 doi: 10.2307/2335991

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., … Oliphant, T. E. (2020, September). Array programming with NumPy. *Nature*, *585*(7825), 357–362. Retrieved 2023-08-09, from https://www.nature.com/articles/s41586-020-2649-2 doi: 10.1038/s41586-020-2649-2

Hideshima, T., Mitsiades, C., Tonon, G., Richardson, P. G., & Anderson, K. C. (2007, August). Understanding multiple myeloma pathogenesis in the bone marrow to identify new therapeutic targets. *Nature Reviews Cancer*, *7*(8), 585–598. Retrieved 2023-02-07, from https://www.nature.com/articles/nrc2189 doi: 10.1038/nrc2189

Hose, D., Rème, T., Hielscher, T., Moreaux, J., Messner, T., Seckinger, A., … Goldschmidt, H. (2011, January). Proliferation is a central independent prognostic factor and target for personalized and risk-adapted treatment in multiple myeloma. *Haematologica*, *96*(1), 87–95. doi: 10.3324/haematol.2010.030296

Hothorn, T., & Lausen, B. (n.d.). *Maximally Selected Rank Statistics in R.* Retrieved from http://cran.r-project.org/web/packages/maxstat/index.html.

Howe, A., & Chain, P. S. G. (2015). Challenges and opportunities in understanding microbial communities with metagenome assembly (accompanied by IPython Notebook tutorial). *Frontiers in Microbiology*, *6*, 678. doi:

10.3389/fmicb.2015.00678

Hu, X., Villodre, E. S., Larson, R., Rahal, O. M., Wang, X., Gong, Y., … Debeb, B. G. (2021, January). Decorin-mediated suppression of tumorigenesis, invasion, and metastasis in inflammatory breast cancer. *Communications Biology*, *4*(1), 72. doi: 10.1038/s42003-020-01590-0

Huang, S.-Y., Lin, H.-H., Yao, M., Tang, J.-L., Wu, S.-J., Hou, H.-A., … Tien, H.-F. (2015). Higher Decorin Levels in Bone Marrow Plasma Are Associated with Superior Treatment Response to Novel Agent-Based Induction in Patients with Newly Diagnosed Myeloma - A Retrospective Study. *PloS One*, *10*(9), e0137552. doi: 10.1371/journal.pone.0137552

Hunter, J. D. (2007, May). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, *9*(3), 90–95. Retrieved 2023-11-15, from https://ieeexplore.ieee.org/document/4160265 doi: 10.1109/MCSE.2007.55

Incerti, D., Thom, H., Baio, G., & Jansen, J. P. (2019, May). R You Still Using Excel? The Advantages of Modern Software Tools for Health Technology Assessment. *Value in Health*, *22*(5), 575–579. Retrieved 2024-03-11, from https://www.sciencedirect.com/science/article/pii/S1098301519300506 doi: 10.1016/j.jval.2019.01.003

Ioannidis, J. P. A. (2005, August). Why Most Published Research Findings Are False. *PLOS Medicine*, *2*(8), e124. Retrieved 2024-04-22, from https://journals.plos.org/plosmedicine/article?id=10.1371/journal.pmed.0020124 doi: 10.1371/journal.pmed.0020124

Jansen, B. J. H., Gilissen, C., Roelofs, H., Schaap-Oziemlak, A., Veltman, J. A., Raymakers, R. A. P., … Adema, G. J. (2010, April). Functional differences between mesenchymal stem cell populations are reflected by their transcriptome. *Stem cells and development*, *19*(4), 481–490. doi: 10.1089/scd.2009.0288

Jung, S.-H., & Lee, J.-J. (2022, April). Update on primary plasma cell leukemia. *Blood Research*, *57*(S1), 62–66. doi: 10.5045/br.2022.2022033

Kaplan, E. L., & Meier, P. (1958, June). Nonparametric Estimation from Incomplete Observations. *Journal of the American Statistical Association*, *53*(282), 457–481. Retrieved 2023-08-07, from http://www.tandfonline.com/doi/abs/10.1080/01621459.1958.10501452 doi: 10.1080/01621459.1958.10501452

Kastritis, E., Moulopoulos, L. A., Terpos, E., Koutoulidis, V., & Dimopoulos, M. A. (2014, December). The prognostic importance of the presence of more than one focal lesion in spine MRI of patients with asymptomatic (smoldering) multiple myeloma. *Leukemia*, *28*(12), 2402–2403. Retrieved 2024-05-23, from https://www.nature.com/articles/leu2014230 doi: 10.1038/leu.2014.230

Katz, B.-Z. (2010, June). Adhesion molecules—The lifelines of multiple myeloma cells. *Seminars in Cancer Biology*, *20*(3), 186–195. Retrieved 2021-07-04, from https://linkinghub.elsevier.com/retrieve/pii/S1044579X10000246 doi: 10.1016/j.semcancer.2010.04.003

Kawano, M. M., Huang, N., Tanaka, H., Ishikawa, H., Sakai, A., Tanabe, O., … Kuramoto, A. (1991, December). Homotypic cell aggregations of human myeloma cells with ICAM-1 and LFA-1 molecules. *British Journal of Haematology*, *79*(4), 583–588. doi: 10.1111/j.1365-2141.1991.tb08085.x

Kazman, R., Bianco, P., Ivers, J., & Klein, J. (2020, December). *Maintainability* (Report). Carnegie Mellon University. Retrieved 2024-03-07, from https://kilthub.cmu.edu/articles/report/Maintainability/12954908/1 doi: 10.1184/R1/12954908.v1

Keats, J. J., Chesi, M., Egan, J. B., Garbitt, V. M., Palmer, S. E., Braggio, E., … Bergsagel, P. L. (2012, August). Clonal competition with alternating dominance in multiple myeloma. *Blood*, *120*(5), 1067–1076. doi: 10.1182/blood-2012-01-405985

Kelleher, R. (2024, January). *NVIDIA CEO: 'This Year, Every Industry Will Become a Technology Industry'.* Retrieved 2024-05-03, from https://blogs.nvidia.com/blog/nvidia-ceo-ai-drug-discovery-jp-morgan-healthcare-2024/

Kelly, B. S., Kirwan, A., Quinn, M. S., Kelly, A. M., Mathur, P., Lawlor, A., & Killeen, R. P. (2023, May). The ethical matrix as a method for involving people living with disease and the wider public (PPI) in near-term artificial intelligence research. *Radiography (London, England: 1995)*, *29 Suppl 1*, S103-S111. doi: 10.1016/j.radi.2023.03.009

Kibler, C., Schermutzki, F., Waller, H. D., Timpl, R., Müller, C. A., & Klein, G. (1998, June). Adhesive interactions of human multiple myeloma cell lines with different extracellular matrix molecules. *Cell Adhesion and Communication*, *5*(4), 307–323. doi: 10.3109/15419069809040300

Kim, D., Langmead, B., & Salzberg, S. L. (2015, April). HISAT: A fast spliced aligner with low memory requirements. *Nature methods*, *12*(4), 357–360. Retrieved 2024-04-26, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4655817/ doi: 10.1038/nmeth.3317

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., … Jupyter Development Team (2016). *Jupyter Notebooks—a publishing format for reproducible computational workflows*. Retrieved 2024-04-20, from https://ui.adsabs.harvard.edu/abs/2016ppap.book...87K doi: 10.3233/978-1-61499-649-1-87

Krekel, H., Oliveira, B., Pfannschmidt, R., Bruynooghe, F., Laugher, B., & Bruhin, F. (2004). *Pytest*. Retrieved from https://github.com/pytest-dev/pytest

Krzywinski, M., & Savig, E. (2013, July). Multidimensional data. *Nature methods*, *10*(7), 595. Retrieved 2024-04-22, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6092027/

Kundu, S., Jha, S. B., Rivera, A. P., Flores Monar, G. V., Islam, H., Puttagunta, S. M., … Sange, I. (2022, February). Multiple Myeloma and Renal Failure: Mechanisms, Diagnosis, and Management. *Cureus*, *14*(2), e22585. Retrieved 2024-05-23, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8958144/ doi: 10.7759/cureus.22585

Kuric, M. (2024, April). *Markur4/plotastic*. Retrieved 2024-05-02, from https://github.com/markur4/plotastic

Kuric, M., Beck, S., Schneider, D., Rindt, W., Evers, M., Meißner-Weigl, J., … Ebert, R. (2024, April). Modeling Myeloma Dissemination In Vitro with hMSC-interacting Subpopulations of INA-6 Cells and Their Aggregation/Detachment Dynamics. *Cancer Research Communications*, *4*(4), 1150–1164. Retrieved 2024-05-14, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11057410/ doi: 10.1158/2767-9764.CRC-23-0411

Kuric, M., & Ebert, R. (2024, March). Plotastic: Bridging Plotting and Statistics in Python. *Journal of Open Source Software*, *9*(95), 6304. Retrieved 2024-03-11, from https://joss.theoj.org/papers/10.21105/joss.06304 doi: 10.21105/joss.06304

Kyle, R. A. (1997, February). Monoclonal gammopathy of undetermined significance and solitary plasmacytoma. Implications for progression to overt multiple myeloma. *Hematology/Oncology Clinics of North America*, *11*(1), 71–87. doi: 10.1016/s0889-8588(05)70416-0

Lai, T.-Y., Cao, J., Ou-Yang, P., Tsai, C.-Y., Lin, C.-W., Chen, C.-C., … Lee, C.-Y. (2022, April). Different methods of detaching adherent cells and their effects on the cell surface expression of Fas receptor and Fas ligand. *Scientific Reports*, *12*(1), 5713. Retrieved 2023-06-01, from https://www.nature.com/articles/s41598-022-09605-y doi: 10.1038/s41598-022-09605-y

Lakhlifi, C., Lejeune, F.-X., Rouault, M., Khamassi, M., & Rohaut, B. (2023, April). Illusion of knowledge in statistics among clinicians: Evaluating the alignment between objective accuracy and subjective confidence, an online survey. *Cognitive Research: Principles and Implications*, *8*, 23. Retrieved 2024-04-24, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10118231/ doi: 10.1186/s41235-023-00474-1

Leek, J. T., & Peng, R. D. (2015, April). Statistics: P values are just the tip of the iceberg. *Nature*, *520*(7549), 612–612. Retrieved 2024-04-22, from https://www.nature.com/articles/520612a doi: 10.1038/520612a

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., … 1000 Genome Project Data Processing

Subgroup (2009, August). The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, *25*(16), 2078–2079. Retrieved 2023-06-09, from https://doi.org/10.1093/bioinformatics/btp352 doi: 10.1093/bioinformatics/btp352

Localio, A. R., Goodman, S. N., Meibohm, A., Cornell, J. E., Stack, C. B., Ross, E. A., & Mulrow, C. D. (2018, June). Statistical Code to Support the Scientific Story. *Annals of Internal Medicine*, *168*(11), 828–829. Retrieved 2024-04-23, from https://www.acpjournals.org/doi/10.7326/M17-3431 doi: 10.7326/M17-3431

Love, M. I., Huber, W., & Anders, S. (2014, December). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, *15*(12), 550. Retrieved 2024-04-26, from https://doi.org/10.1186/s13059-014-0550-8 doi: 10.1186/s13059-014-0550-8

Mai, E. K., Hielscher, T., Kloth, J. K., Merz, M., Shah, S., Raab, M. S., … Hillengass, J. (2015, June). A magnetic resonance imaging-based prognostic scoring system to predict outcome in transplant-eligible patients with multiple myeloma. *Haematologica*, *100*(6), 818–825. Retrieved 2024-05-23, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4450628/ doi: 10.3324/haematol.2015.124115

Maichl, D. S., Kirner, J. A., Beck, S., Cheng, W.-H., Krug, M., Kuric, M., … Jundt, F. (2023, September). Identification of NOTCH-driven matrisome-associated genes as prognostic indicators of multiple myeloma patient survival. *Blood Cancer Journal*, *13*(1), 1–6. Retrieved 2023-09-05, from https://www.nature.com/articles/s41408-023-00907-6 doi: 10.1038/s41408-023-00907-6

Majithia, N., Rajkumar, SV., Lacy, MQ., Buadi, FK., Dispenzieri, A., Gertz, MA., … Kumar, SK. (2016, November). Early relapse following initial therapy for multiple myeloma predicts poor outcomes in the era of novel agents. *Leukemia*, *30*(11), 2208–2213. Retrieved 2022-10-15, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5541860/ doi: 10.1038/leu.2016.147

Mangolini, M., & Ringshausen, I. (2020, February). Bone Marrow Stromal Cells Drive Key Hallmarks of B Cell Malignancies. *International Journal of Molecular Sciences*, *21*(4), 1466. Retrieved 2023-05-02, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7073037/ doi: 10.3390/ijms21041466

Manifesto for Agile Software Development. (2001). Retrieved 2024-05-14, from http://agilemanifesto.org/

Mateos María-Victoria, Hernández Miguel-Teodoro, Giraldo Pilar, de la Rubia Javier, de Arriba Felipe, Corral Lucía López, … San Miguel Jesús-F. (2013). Lenalidomide plus Dexamethasone for High-Risk Smoldering Multiple Myeloma. *New England Journal of Medicine*, *369*(5), 438–447. Retrieved 2024-05-22, from https://www.nejm.org/doi/full/10.1056/NEJMoa1300439 doi: 10.1056/NEJMoa1300439

McCall, M. N., McMurray, H. R., Land, H., & Almudevar, A. (2014, August). On non-detects in qPCR data. *Bioinformatics*, *30*(16), 2310–2316. Retrieved 2023-04-25, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4133581/ doi: 10.1093/bioinformatics/btu239

McKay, B. S., Irving, P. E., Skumatz, C. M., & Burke, J. M. (1997, November). Cell-cell adhesion molecules and the development of an epithelial phenotype in cultured human retinal pigment epithelial cells. *Experimental Eye Research*, *65*(5), 661–671. doi: 10.1006/exer.1997.0374

McKinney, W. (2010, January). Data Structures for Statistical Computing in Python. In (pp. 56–61). doi: 10.25080/Majora-92bf1922-00a

Mckinney, W. (2011, January). Pandas: A Foundational Python Library for Data Analysis and Statistics. *Python High Performance Science Computer*.

Mesirov, J. P. (2010, January). Accessible Reproducible Research. *Science*, *327*(5964), 415–416. Retrieved 2024-04-22, from https://www.science.org/doi/10.1126/science.1179653 doi: 10.1126/science.1179653

Moleiro, A. F., Conceição, G., Leite-Moreira, A. F., & Rocha-Sousa, A. (2017). A Critical Analysis of the Available In Vitro and Ex Vivo Methods to Study Retinal Angiogenesis. *Journal of Ophthalmology*, *2017*, 3034953. doi: 10.1155/2017/3034953

Moran, M. (2003). Arguments for rejecting the sequential Bonferroni in ecological studies. *Oikos*, *100*(2), 403–

405. Retrieved 2024-04-24, from https://onlinelibrary.wiley.com/doi/abs/10.1034/j.1600-0706.2003.12010.x doi: 10.1034/j.1600-0706.2003.12010.x

Morè, S., Corvatta, L., Manieri, V. M., Morsia, E., Poloni, A., & Offidani, M. (2023, November). Novel Immunotherapies and Combinations: The Future Landscape of Multiple Myeloma Treatment. *Pharmaceuticals*, *16*(11), 1628. Retrieved 2024-05-22, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10675193/ doi: 10.3390/ph16111628

Motulsky, H. (2018). *Intuitive Biostatistics: A Nonmathematical Guide to Statistical Thinking*. Oxford University Press.

Mrozik, K. M., Cheong, C. M., Hewett, D., Chow, A. W., Blaschuk, O. W., Zannettino, A. C., & Vandyke, K. (2015). Therapeutic targeting of N-cadherin is an effective treatment for multiple myeloma. *British Journal of Haematology*, *171*(3), 387–399. Retrieved 2024-05-26, from https://onlinelibrary.wiley.com/doi/abs/10.1111/bjh.13596 doi: 10.1111/bjh.13596

Muruganandan, S., Roman, A. A., & Sinal, C. J. (2009, January). Adipocyte differentiation of bone marrow-derived mesenchymal stem cells: Cross talk with the osteoblastogenic program. *Cellular and molecular life sciences : CMLS*, *66*(2), 236–253. doi: 10.1007/s00018-008-8429-z

Myers, G. J., Sandler, C., & Badgett, T. (2011). *The art of software testing* (3rd ed.). Wiley Publishing. Retrieved from https://malenezi.github.io/malenezi/SE401/Books/114-the-art-of-software-testing-3-edition.pdf

Narzt, W., Pichler, J., Pirklbauer, K., & Zwinz, M. (1998, January). A Reusability Concept for Process Automation Software..

Newville, M., Stensitzki, T., Allen, D. B., & Ingargiola, A. (2014, September). *LMFIT: Non-Linear Least-Square Minimization and Curve-Fitting for Python.* Zenodo. Retrieved 2023-05-30, from https://zenodo.org/record/11813 doi: 10.5281/zenodo.11813

Nilsson, K., Bennich, H., Johansson, S. G., & Pontén, J. (1970, October). Established immunoglobulin producing myeloma (IgE) and lymphoblastoid (IgG) cell lines from an IgE myeloma patient. *Clinical and Experimental Immunology*, *7*(4), 477–489.

Nowotschin, S., & Hadjantonakis, A.-K. (2010, August). Cellular dynamics in the early mouse embryo: From axis formation to gastrulation. *Current opinion in genetics & development*, *20*(4), 420–427. doi: 10.1016/j.gde.2010.05.008

Ocias, L. F., Larsen, T. S., Vestergaard, H., Friis, L. S., Abildgaard, N., Frederiksen, H., & Academy of Geriatric Cancer Research (AgeCare). (2016). Trends in hematological cancer in the elderly in Denmark, 1980-2012. *Acta Oncologica (Stockholm, Sweden)*, *55 Suppl 1*, 98–107. doi: 10.3109/0284186X.2015.1115124

O'Connor, B. P., Raman, V. S., Erickson, L. D., Cook, W. J., Weaver, L. K., Ahonen, C., … Noelle, R. J. (2004, January). BCMA Is Essential for the Survival of Long-lived Bone Marrow Plasma Cells. *The Journal of Experimental Medicine*, *199*(1), 91–98. Retrieved 2024-05-26, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1887725/ doi: 10.1084/jem.20031330

Okuno, Y., Takahashi, T., Suzuki, A., Ichiba, S., Nakamura, K., Okada, T., … Imura, H. (1991, February). In vitro growth pattern of myeloma cells in liquid suspension or semi-solid culture containing interleukin-6. *International Journal of Hematology*, *54*(1), 41–47.

Ordak, M. (2023, September). ChatGPT's Skills in Statistical Analysis Using the Example of Allergology: Do We Have Reason for Concern? *Healthcare (Basel, Switzerland)*, *11*(18), 2554. doi: 10.3390/healthcare11182554

Paiva, B., Paino, T., Sayagues, J.-M., Garayoa, M., San-Segundo, L., Martín, M., … San Miguel, J. F. (2013, November). Detailed characterization of multiple myeloma circulating tumor cells shows unique phenotypic, cytogenetic, functional, and circadian distribution profile. *Blood*, *122*(22), 3591–3598. doi: 10.1182/blood-2013-06-510453

Paiva, B., Pérez-Andrés, M., Vídriales, M.-B., Almeida, J., de las Heras, N., Mateos, M.-V., … Myeloma Stem Cell Network (MSCNET) (2011, April). Competition between clonal plasma cells and normal cells for potentially overlapping bone marrow niches is associated with a progressively altered cellular distribution in MGUS vs myeloma. *Leukemia*, *25*(4), 697–706. doi: 10.1038/leu.2010.320

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., … Chintala, S. (2019, December). *PyTorch: An Imperative Style, High-Performance Deep Learning Library* (No. arXiv:1912.01703). arXiv. Retrieved 2024-03-07, from http://arxiv.org/abs/1912.01703 doi: 10.48550/arXiv.1912.01703

Peng, R. D. (2011, December). Reproducible Research in Computational Science. *Science*, *334*(6060), 1226–1227. Retrieved 2024-03-18, from https://www.science.org/doi/10.1126/science.1213847 doi: 10.1126/science.1213847

Perez, F., & Granger, B. E. (2007, May). IPython: A System for Interactive Scientific Computing. *Computing in Science & Engineering*, *9*(3), 21–29. Retrieved 2024-04-20, from https://ieeexplore.ieee.org/document/4160251 doi: 10.1109/MCSE.2007.53

Perneger, T. V. (1998, April). What's wrong with Bonferroni adjustments. *BMJ : British Medical Journal*, *316*(7139), 1236–1238. Retrieved 2021-11-24, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1112991/

Pfaffl, M. W. (2001, May). A new mathematical model for relative quantification in real-time RT–PCR. *Nucleic Acids Research*, *29*(9), e45. Retrieved 2024-05-16, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC55695/

Pittenger, M. F., Mackay, A. M., Beck, S. C., Jaiswal, R. K., Douglas, R., Mosca, J. D., … Marshak, D. R. (1999). Multilineage Potential of Adult Human Mesenchymal Stem Cells. , *284*(April), 143–148. doi: 10.1126/science.284.5411.143

Podar, K., & Leleu, X. (2021, October). Relapsed/Refractory Multiple Myeloma in 2020/2021 and Beyond. *Cancers*, *13*(20), 5154. Retrieved 2024-05-22, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8534171/ doi: 10.3390/cancers13205154

Polager, S., & Ginsberg, D. (2009, October). P53 and E2f: Partners in life and death. *Nature Reviews Cancer*, *9*(10), 738–748. Retrieved 2023-02-14, from https://www.nature.com/articles/nrc2718 doi: 10.1038/nrc2718

Purschke, M., Rubio, N., Held, K. D., & Redmond, R. W. (2010, November). Phototoxicity of Hoechst 33342 in time-lapse fluorescence microscopy. *Photochemical & Photobiological Sciences*, *9*(12), 1634–1639. Retrieved 2022-03-03, from https://pubs.rsc.org/en/content/articlelanding/2010/pp/c0pp00234h doi: 10.1039/C0PP00234H

PyMOL. (2024). Retrieved 2024-04-30, from https://pymol.org/

The Python Language Reference. (2024). Retrieved 2024-03-07, from https://docs.python.org/3/reference/index.html

Qasim, W., Zhan, H., Samarasinghe, S., Adams, S., Amrolia, P., Stafford, S., … Veys, P. (2017, January). Molecular remission of infant B-ALL after infusion of universal TALEN gene-edited CAR T cells. *Science Translational Medicine*, *9*(374), eaaj2013. doi: 10.1126/scitranslmed.aaj2013

Quanbeck, A., Hennessy, R. G., & Park, L. (2022, November). Applying concepts from "rapid" and "agile" implementation to advance implementation research. *Implementation Science Communications*, *3*(1), 118. doi: 10.1186/s43058-022-00366-3

Qureshi, R., Shaughnessy, D., Gill, K. A. R., Robinson, K. A., Li, T., & Agai, E. (2023, April). Are ChatGPT and large language models "the answer" to bringing us closer to systematic review automation? *Systematic Reviews*, *12*, 72. Retrieved 2024-05-03, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10148473/ doi: 10.1186/s13643-023-02243-z

R Core Team. (2018). *R: A language and environment for statistical computing* [Manual]. Vienna, Austria. Retrieved from https://www.R-project.org/

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners.. Retrieved 2024-03-07, from https://www.semanticscholar.org/paper/Language-Models-are-Unsupervised-Multitask-Learners-Radford-Wu/9405cc0d6169988371b2755e573cc28650d14dfe

Rajkumar, S. V., Dimopoulos, M. A., Palumbo, A., Blade, J., Merlini, G., Mateos, M.-V., … Miguel, J. F. S. (2014, November). International Myeloma Working Group updated criteria for the diagnosis of multiple myeloma. *The Lancet. Oncology*, *15*(12), e538-548. doi: 10.1016/S1470-2045(14)70442-5

Rajkumar, S. V., & Kumar, S. (2020, September). Multiple myeloma current treatment algorithms. *Blood Cancer Journal*, *10*(9), 94. Retrieved 2023-07-03, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7523011/ doi: 10.1038/s41408-020-00359-2

Ramakers, C., Ruijter, J. M., Deprez, R. H., & Moorman, A. F. (2003, March). Assumption-free analysis of quantitative real-time polymerase chain reaction (PCR) data. *Neuroscience Letters*, *339*(1), 62–66. Retrieved 2022-11-27, from https://linkinghub.elsevier.com/retrieve/pii/S0304394002014234 doi: 10.1016/S0304-3940(02)01423-4

Rayhan, A., & Gross, D. (2023). *The Rise of Python: A Survey of Recent Research.* doi: 10.13140/RG.2.2.27388.92809

Read the Docs. (2024). Retrieved 2024-05-03, from https://docs.readthedocs.io/en/stable/index.html

Rebl, H., Finke, B., Schroeder, K., & Nebe, J. B. (2010, October). Time-dependent metabolic activity and adhesion of human osteoblast-like cells on sensor chips with a plasma polymer nanolayer. *The International Journal of Artificial Organs*, *33*(10), 738–748.

Ribatti, D., Tamma, R., & Annese, T. (2020, June). Epithelial-Mesenchymal Transition in Cancer: A Historical Overview. *Translational Oncology*, *13*(6), 100773. doi: 10.1016/j.tranon.2020.100773

Rigsby, R. E., & Parker, A. B. (2016, September). Using the PyMOL application to reinforce visual understanding of protein structure. *Biochemistry and Molecular Biology Education: A Bimonthly Publication of the International Union of Biochemistry and Molecular Biology*, *44*(5), 433–437. doi: 10.1002/bmb.20966

Robinson, M. D., McCarthy, D. J., & Smyth, G. K. (2010, January). edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics (Oxford, England)*, *26*(1), 139–140. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/19910308 doi: 10.1093/bioinformatics/btp616

Rueden, C. T., Schindelin, J., Hiner, M. C., DeZonia, B. E., Walter, A. E., Arena, E. T., & Eliceiri, K. W. (2017, November). ImageJ2: ImageJ for the next generation of scientific image data. *BMC Bioinformatics*, *18*(1), 529. Retrieved 2024-04-25, from https://doi.org/10.1186/s12859-017-1934-z doi: 10.1186/s12859-017-1934-z

Ruijter, J. M., Barnewall, R. J., Marsh, I. B., Szentirmay, A. N., Quinn, J. C., van Houdt, R., … van den Hoff, M. J. B. (2021, June). Efficiency Correction Is Required for Accurate Quantitative PCR Analysis and Reporting. *Clinical Chemistry*, *67*(6), 829–842. Retrieved 2023-05-27, from https://doi.org/10.1093/clinchem/hvab052 doi: 10.1093/clinchem/hvab052

Ruiz-Villalba, A., Ruijter, J. M., & van den Hoff, M. J. B. (2021, May). Use and Misuse of Cq in qPCR Data Analysis and Reporting. *Life*, *11*(6), 496. Retrieved 2023-04-25, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8229287/ doi: 10.3390/life11060496

Ruksakulpiwat, S., Kumar, A., & Ajibade, A. (2023, May). Using ChatGPT in Medical Research: Current Status and Future Directions. *Journal of Multidisciplinary Healthcare*, *16*, 1513–1520. Retrieved 2024-05-03, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10239248/ doi: 10.2147/JMDH.S413470

Sacchetti, B., Funari, A., Remoli, C., Giannicola, G., Kogler, G., Liedtke, S., … Bianco, P. (2016). No identical "mesenchymal stem cells" at different times and sites: Human committed progenitors of distinct origin and

differentiation potential are incorporated as adventitial cells in microvessels. *Stem Cell Reports*, *6*(6), 897–913. Retrieved from http://dx.doi.org/10.1016/j.stemcr.2016.05.011 doi: 10.1016/j.stemcr.2016.05.011

Sandve, G. K., Nekrutenko, A., Taylor, J., & Hovig, E. (2013, October). Ten Simple Rules for Reproducible Computational Research. *PLoS Computational Biology*, *9*(10), e1003285. Retrieved 2024-03-07, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3812051/ doi: 10.1371/journal.pcbi.1003285

Santos, B. S., Silva, I., Ribeiro-Dantas, M. d. C., Alves, G., Endo, P. T., & Lima, L. (2020, October). COVID-19: A scholarly production dataset report for research analysis. *Data in Brief*, *32*, 106178. doi: 10.1016/j.dib.2020.106178

Sanz-Rodríguez, F., Ruiz-Velasco, N., Pascual-Salcedo, D., & Teixidó, J. (1999, December). Characterization of VLA-4-dependent myeloma cell adhesion to fibronectin and VCAM-1: VLA-4-dependent Myeloma Cell Adhesion. *British Journal of Haematology*, *107*(4), 825–834. Retrieved 2023-04-02, from http://doi.wiley.com/10.1046/j.1365-2141.1999.01762.x doi: 10.1046/j.1365-2141.1999.01762.x

Sarin, V., Yu, K., Ferguson, I. D., Gugliemini, O., Nix, M. A., Hann, B., … Wiita, A. P. (2020, October). Evaluating the efficacy of multiple myeloma cell lines as models for patient tumors via transcriptomic correlation analysis. *Leukemia*, *34*(10), 2754–2765. doi: 10.1038/s41375-020-0785-1

Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and Statistical Modeling with Python. In *Python in Science Conference* (pp. 92–96). Austin, Texas. Retrieved 2023-05-29, from https://conference.scipy.org/proceedings/scipy2010/seabold.html doi: 10.25080/Majora-92bf1922-011

Seckinger, A., Delgado, J. A., Moser, S., Moreno, L., Neuber, B., Grab, A., … Vu, M. D. (2017, March). Target Expression, Generation, Preclinical Activity, and Pharmacokinetics of the BCMA-T Cell Bispecific Antibody EM801 for Multiple Myeloma Treatment. *Cancer Cell*, *31*(3), 396–410. Retrieved 2023-07-21, from https://www.cell.com/cancer-cell/abstract/S1535-6108(17)30016-8 doi: 10.1016/j.ccell.2017.02.002

Seckinger, A., Hillengass, J., Emde, M., Beck, S., Kimmich, C., Dittrich, T., … Hose, D. (2018). CD38 as Immunotherapeutic Target in Light Chain Amyloidosis and Multiple Myeloma-Association With Molecular Entities, Risk, Survival, and Mechanisms of Upfront Resistance. *Frontiers in Immunology*, *9*, 1676. doi: 10.3389/fimmu.2018.01676

Shenghui, H., Nakada, D., & Morrison, S. J. (2009). Mechanisms of Stem Cell Self-Renewal. *Annual Review of Cell and Developmental Biology*, *25*(1), 377–406. Retrieved from https://doi.org/10.1146/annurev.cellbio.042308.113248 doi: 10.1146/annurev.cellbio.042308.113248

Sherina, V. (2020). Multiple imputation and direct estimation for qPCR data with non-detects.

Siegel, R. L., Giaquinto, A. N., & Jemal, A. (2024). Cancer statistics, 2024. *CA: A Cancer Journal for Clinicians*, *74*(1), 12–49. Retrieved 2024-05-21, from https://onlinelibrary.wiley.com/doi/abs/10.3322/caac.21820 doi: 10.3322/caac.21820

Smith, A. M., Niemeyer, K. E., Katz, D. S., Barba, L. A., Githinji, G., Gymrek, M., … Vanderplas, J. T. (2018). Journal of Open Source Software (JOSS): Design and first-year review. *PeerJ Preprints*, *4*, e147. doi: 10.7717/peerj-cs.147

Solimando, A. G., Malerba, E., Leone, P., Prete, M., Terragna, C., Cavo, M., & Racanelli, V. (2022, September). Drug resistance in multiple myeloma: Soldiers and weapons in the bone marrow niche. *Frontiers in Oncology*, *12*, 973836. Retrieved 2022-10-23, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9533079/ doi: 10.3389/fonc.2022.973836

Sphinx. (2024). Retrieved 2024-05-03, from https://docs.readthedocs.io/en/stable/intro/getting-started-with-sphinx.html

Sprynski, A. C., Hose, D., Caillot, L., Rème, T., Shaughnessy, J. D., Barlogie, B., … Klein, B. (2009, May). The role of IGF-1 as a major growth factor for myeloma cell lines and the prognostic relevance of the expression of its receptor. *Blood*, *113*(19), 4614–4626. Retrieved 2023-06-29, from https://www.ncbi.nlm.nih.gov/pmc/

articles/PMC2691749/ doi: 10.1182/blood-2008-07-170464

Standal, T., Seidel, C., Plesner, T., Sanderson, R., Waage, A., Børset, M., & Sundan, A. (2002, November). Osteoprotegerin is bound, internalized, and degraded by multiple myeloma cells. *Blood*, *100*, 3002–7. doi: 10.1182/blood-2002-04-1190

Stock, P., Bruckner, S., Winkler, S., Dollinger, M. M., & Christ, B. (2014, April). Human bone marrow mesenchymal stem cell-derived hepatocytes improve the mouse liver after acute acetaminophen intoxication by preventing progress of injury. *International journal of molecular sciences*, *15*(4), 7004–7028. doi: 10.3390/ijms15047004

Sullivan, G. M., & Feinn, R. S. (2021, August). Facts and Fictions About Handling Multiple Comparisons. *Journal of Graduate Medical Education*, *13*(4), 457–460. Retrieved 2024-03-10, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8370375/ doi: 10.4300/JGME-D-21-00599.1

Tabolacci, C., De Martino, A., Mischiati, C., Feriotto, G., & Beninati, S. (2019, January). The Role of Tissue Transglutaminase in Cancer Cell Initiation, Survival and Progression. *Medical Sciences*, *7*(2), 19. Retrieved 2023-03-17, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6409630/ doi: 10.3390/medsci7020019

Tai, Y.-T., Li, X.-F., Breitkreutz, I., Song, W., Neri, P., Catley, L., … Anderson, K. C. (2006, July). Role of B-cell-activating factor in adhesion and growth of human multiple myeloma cells in the bone marrow microenvironment. *Cancer Research*, *66*(13), 6675–6682. doi: 10.1158/0008-5472.CAN-06-0190

Tam, P. P., & Beddington, R. S. (1987, January). The formation of mesodermal tissues in the mouse embryo during gastrulation and early organogenesis. *Development (Cambridge, England)*, *99*(1), 109–126.

Taskiran, I. I., Spanier, K. I., Dickmänken, H., Kempynck, N., Pančíková, A., Ekşi, E. C., … Aerts, S. (2024, February). Cell-type-directed design of synthetic enhancers. *Nature*, *626*(7997), 212–220. Retrieved 2024-04-21, from https://www.nature.com/articles/s41586-023-06936-2 doi: 10.1038/s41586-023-06936-2

Team, T. P. D. (2020, February). *Pandas-dev/pandas: Pandas.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.3509134 doi: 10.5281/zenodo.3509134

Terpos, E., Migkou, M., Christoulas, D., Gavriatopoulou, M., Eleutherakis-Papaiakovou, E., Kanellias, N., … Dimopoulos, M. A. (2016, May). Increased circulating VCAM-1 correlates with advanced disease and poor survival in patients with multiple myeloma: Reduction by post-bortezomib and lenalidomide treatment. *Blood Cancer Journal*, *6*(5), e428. Retrieved 2021-02-03, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4916305/ doi: 10.1038/bcj.2016.37

Terpos, E., Ntanasis-Stathopoulos, I., Gavriatopoulou, M., & Dimopoulos, M. A. (2018, January). Pathogenesis of bone disease in multiple myeloma: From bench to bedside. *Blood Cancer Journal*, *8*(1), 7. doi: 10.1038/s41408-017-0037-4

Thompson, S., Dowrick, T., Ahmad, M., Xiao, G., Koo, B., Bonmati, E., … Clarkson, M. J. (2020, July). SciKit-Surgery: Compact libraries for surgical navigation. *International Journal of Computer Assisted Radiology and Surgery*, *15*(7), 1075–1084. doi: 10.1007/s11548-020-02180-5

Thumallapally, N., Meshref, A., Mousa, M., & Terjanian, T. (2017, January). Solitary plasmacytoma: Population-based analysis of survival trends and effect of various treatment modalities in the USA. *BMC Cancer*, *17*, 13. Retrieved 2024-05-21, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5216567/ doi: 10.1186/s12885-016-3015-5

Trapnell, C., Roberts, A., Goff, L., Pertea, G., Kim, D., Kelley, D. R., … Pachter, L. (2012, March). Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nature Protocols*, *7*(3), 562–578. doi: 10.1038/nprot.2012.016

Turesson, I., Bjorkholm, M., Blimark, C. H., Kristinsson, S., Velez, R., & Landgren, O. (2018, April). Rapidly changing myeloma epidemiology in the general population: Increased incidence, older patients, and longer survival. *European journal of haematology*, 10.1111/ejh.13083. Retrieved 2024-05-22, from https://www.ncbi

.nlm.nih.gov/pmc/articles/PMC6195866/ doi: 10.1111/ejh.13083

Two new interleukin-6 dependent plasma cell lines carrying a chromosomal abnormality involving the IL-6 gene locus.Abstract Two plasma cell lines, INA-6 and JK-6, have been initiated and continuously cultured from two patients with malignant plasma cell diseases. Both cell lines are EBNA negative and show morphological and immunophenotypical features of plasma cells. INA-6 expresses the CD39 and CDw75 antigens, JK-6 is strongty positive with CD38 and CD39 antibodies. By flow cytometry they were non-reactive with Ia antibodies and B ceil reagents CD19, CD20, CD21, CD22, and CD24. While INA-6 cells are releasing kappa light chains only, JK-6 cells produce IgG kappa. Both cell lines could only be initiated with IL-6 supplemented medium and remained IL-6 responsive throughout continuous culture. INA-6 is strictly dependent on IL-6. No spontaneously secreted IL-6 was found nor could it be induced by IL-1beta /TNFalpha stimulation. Molecular analysis with RT-PCR revealed mRNA for the IL-6 receptor in both lines. No IL-6 mRNA was detectable in INA-6 cells, while in JK-6 minute amounts were observed. Cytogenetic analysis of both lines revealed, among other abnormalities, a deletion (7)(p13). lnterestingly, the 7p deletion affects the location ot the IL-6 gene. ln both cell lines, IL-6 dependent proliferation could be inhibited by IFNalpha. IFNalpha had growth regulatory effects only on JK-6: While high concentrations were inhibitory, low IFNalpha amounts were clearly stimulatory. A wide variety of other cytokines including GM-CSF and IL-11 did not have the capacity to influence proliferation. These plasma cell lines do not only allow to turther characterize regulatory events in plasma cell neoplasias but also provide tools to study therapeutic interventions. (n.d.). Retrieved 2023-03-22, from https://www.cellosaurus.org/cellopub/CLPUB00060

Ullah, I., Subbarao, R. B., & Rho, G. J. (2015). Human mesenchymal stem cells - current trends and future prospective Bioscience Reports.
doi: 10.1042/BSR20150025

Urashima, M., Chauhan, D., Uchiyama, H., Freeman, G., & Anderson, K. (1995, April). CD40 ligand triggered interleukin-6 secretion in multiple myeloma. *Blood*, *85*(7), 1903–1912. Retrieved 2021-02-01, from https://ashpublications.org/blood/article/85/7/1903/123565/CD40-ligand-triggered-interleukin6-secretion-in doi: 10.1182/blood.V85.7.1903.bloodjournal8571903

Vallat, R. (2018, November). Pingouin: Statistics in Python. *Journal of Open Source Software*, *3*(31), 1026. Retrieved 2023-05-29, from https://joss.theoj.org/papers/10.21105/joss.01026 doi: 10.21105/joss.01026

van Rossum, G., Lehtosalo, J., & Langa, L. (2014). *PEP 484 – Type Hints | peps.python.org.* Retrieved 2024-03-08, from https://peps.python.org/pep-0484/

Vande Broek, I., Vanderkerken, K., Van Camp, B., & Van Riet, I. (2008). Extravasation and homing mechanisms in multiple myeloma. *Clinical & Experimental Metastasis*, *25*(4), 325–334. doi: 10.1007/s10585-007-9108-4

Van Valckenborgh, E., Croucher, P. I., De Raeve, H., Carron, C., De Leenheer, E., Blacher, S., … Vanderkerken, K. (2004, September). Multifunctional role of matrix metalloproteinases in multiple myeloma: A study in the 5T2MM mouse model. *The American Journal of Pathology*, *165*(3), 869–878. doi: 10.1016/S0002-9440(10)63349-4

Viguet-Carrin, S., Garnero, P., & Delmas, P. D. (2006, March). The role of collagen in bone strength. *Osteoporosis International*, *17*(3), 319–336. Retrieved 2023-12-20, from https://doi.org/10.1007/s00198-005-2035-9 doi: 10.1007/s00198-005-2035-9

Wadgaonkar, R., Phelps, K. M., Haque, Z., Williams, A. J., Silverman, E. S., & Collins, T. (1999, January). CREB-binding protein is a nuclear integrator of nuclear factor-kappaB and p53 signaling. *The Journal of Biological Chemistry*, *274*(4), 1879–1882. doi: 10.1074/jbc.274.4.1879

Waskom, M. L. (2021, April). Seaborn: Statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. Retrieved 2023-03-26, from https://joss.theoj.org/papers/10.21105/joss.03021 doi: 10.21105/joss.03021

Webster, G. A., & Perkins, N. D. (1999, May). Transcriptional Cross Talk between NF-κB and p53. *Molecular and Cellular Biology*, *19*(5), 3485–3495. Retrieved 2023-07-04, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC84141/

Weetall, M., Hugo, R., Maida, S., West, S., Wattanasin, S., Bouhel, R., … Friedman, C. (2001, June). A Homogeneous Fluorometric Assay for Measuring Cell Adhesion to Immobilized Ligand Using V-Well Microtiter Plates. *Analytical Biochemistry*, *293*(2), 277–287. Retrieved 2022-09-25, from https://linkinghub.elsevier.com/retrieve/pii/S0003269701951401 doi: 10.1006/abio.2001.5140

Weiss, C. J. (2022, September). Visualizing protein big data using Python and Jupyter notebooks. *Biochemistry and Molecular Biology Education: A Bimonthly Publication of the International Union of Biochemistry and Molecular Biology*, *50*(5), 431–436. doi: 10.1002/bmb.21621

West, K. (2018, July). *Reinventing Research: Agile in the Academic Laboratory | Agile Alliance.* Retrieved 2024-05-14, from https://www.agilealliance.org/resources/experience-reports/reinventing-research-agile-in-the-academic-laboratory/

Wickham, H. (2014, September). Tidy Data. *Journal of Statistical Software*, *59*, 1–23. Retrieved 2023-11-15, from https://doi.org/10.18637/jss.v059.i10 doi: 10.18637/jss.v059.i10

Wilkins, A., Kemp, K., Ginty, M., Hares, K., Mallam, E., & Scolding, N. (2009, July). Human bone marrow-derived mesenchymal stem cells secrete brain-derived neurotrophic factor which promotes neuronal survival in vitro. *Stem cell research*, *3*(1), 63–70. doi: 10.1016/j.scr.2009.02.006

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., … Mons, B. (2016, March). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, *3*(1), 160018. Retrieved 2024-03-18, from https://www.nature.com/articles/sdata201618 doi: 10.1038/sdata.2016.18

Witwer, K. W. (2013, February). Data submission and quality in microarray-based microRNA profiling. *Clinical chemistry*, *59*(2), 392–400. Retrieved 2024-04-22, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4037921/ doi: 10.1373/clinchem.2012.193813

Wong, A. D., & Searson, P. C. (2017, November). Mitosis-mediated intravasation in a tissue-engineered tumor-microvessel platform. *Cancer research*, *77*(22), 6453–6461. Retrieved 2023-07-14, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5690825/ doi: 10.1158/0008-5472.CAN-16-3279

Xu, W., Zhang, X., Qian, H., Zhu, W., Sun, X., Hu, J., … Chen, Y. (2004, July). Mesenchymal stem cells from adult human bone marrow differentiate into a cardiomyocyte phenotype in vitro. *Experimental biology and medicine (Maywood, N.J.)*, *229*(7), 623–631.

Yang, A., Troup, M., & Ho, J. W. (2017, July). Scalability and Validation of Big Data Bioinformatics Software. *Computational and Structural Biotechnology Journal*, *15*, 379–386. Retrieved 2024-03-07, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5537105/ doi: 10.1016/j.csbj.2017.07.002

Yang, P., Qu, Y., Wang, M., Chu, B., Chen, W., Zheng, Y., … Qian, Z. (2022, June). Pathogenesis and treatment of multiple myeloma. *MedComm*, *3*(2), e146. Retrieved 2024-05-21, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9162151/ doi: 10.1002/mco2.146

Zeissig, M. N., Zannettino, A. C. W., & Vandyke, K. (2020, December). Tumour Dissemination in Multiple Myeloma Disease Progression and Relapse: A Potential Therapeutic Target in High-Risk Myeloma. *Cancers*, *12*(12). Retrieved 2021-02-03, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7761917/ doi: 10.3390/cancers12123643

Zerbino, D. R., Achuthan, P., Akanni, W., Amode, M. R., Barrell, D., Bhai, J., … Flicek, P. (2018, January). Ensembl 2018. *Nucleic Acids Research*, *46*(D1), D754-D761. Retrieved 2023-05-27, from https://doi.org/10.1093/nar/gkx1098 doi: 10.1093/nar/gkx1098

Zhou, F., Meng, S., Song, H., & Claret, F. X. (2013, November). Dickkopf-1 is a key regulator of myeloma bone

disease: Opportunities and challenges for therapeutic intervention. *Blood reviews*, *27*(6), 261–267. Retrieved 2023-02-18, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4133945/ doi: 10.1016/j.blre.2013.08.002

Zhou, Y., Zhou, B., Pache, L., Chang, M., Khodabakhshi, A. H., Tanaseichuk, O., … Chanda, S. K. (2019, April). Metascape provides a biologist-oriented resource for the analysis of systems-level datasets. *Nature Communications*, *10*(1), 1523. Retrieved 2023-02-09, from https://www.nature.com/articles/s41467-019-09234-6 doi: 10.1038/s41467-019-09234-6

Ziemann, M., Eren, Y., & El-Osta, A. (2016, August). Gene name errors are widespread in the scientific literature. *Genome Biology*, *17*(1), 177. Retrieved 2024-04-30, from https://doi.org/10.1186/s13059-016-1044-7 doi: 10.1186/s13059-016-1044-7

Appendices

A Supplementary Data & Methods

A.1 Figures

A.2 Tables

A.3 Materials & Methods

B Documentation of plotastic