D Summary

# CONTEXT:

I am currently writing my PhD Thesis with the title:

> Development and Semi-Automated Analysis of an \textit{in vitro} Dissemination Model \\
> for Myeloma Cells Interacting with Mesenchymal Stromal Cells

Please notice that I am using latex code, I want you to understand the commands I am using and also output suggestions in latex code, adhering to my style and using commands correctly.

My thesis has two chapters: Chapter 1 focuses on lab-work in biomedicine, describing new in vitro methods for describing the dissemination of multiple myeloma, and is published as "Modeling Myeloma Dissemination In Vitro with hMSC-interacting Subpopulations of INA-6 Cells and Their Aggregation/Detachment Dynamics" in cancer research communications. Here is the abstract of the paper for context:

Chapter 2 is a python software I developed called plotastic. It's supposed to make data analysis of large multidimensional datasets easier.

My thesis contains a summarising discussion where I outlay unpublished challenges for chapter 1 and 2, but also combine chapter 1 and 2. Note that the paragraphs for the cancer part of the summarising discussion is not yet finished, though raw brainstorming can give you ideas where it will be headed.

The thesis also requires a one-page summary, that still needs to be written.

# REQUEST

I am going to provide you with parts of my introduction, abstracts of both chapters, their discussions, and the current draft of the summarising discussion. Please write a summary for my complete thesis. Make it easy to understand, but be concise. Use latex code and use a style thats similar to mine.

# PARTS OF THE INTRODUCTION

\unnsubsection{Multidimensional Data in Biomedical Research}%
\label{sec:intro_multidimensional_data}%
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
\cite{reblTimedependentMetabolicActivity2010, mckayCellcellAdhesionMolecules1997,bolado-carrancioPeriodicPropagatingWaves2020a}.

Multidimensional data encompass datasets where multiple \emph{independent
variables} (here reffered to as \emph{factors}) can influence one
\emph{dependent variables} (\emph{outcomes}) \cite{krzywinskiMultidimensionalData2013}.
% Statistically speaking, an experiment assesses if an \emph{independent variable}
% (here reffered to as \emph{factors}) has an influence on a \emph{dependent
% variable} (\emph{outcome}), e.g. ``**Does \emph{Time} influence \emph{Gene**
`% Expression}?'' \cite{motulskyIntuitiveBiostatisticsNonmathematical2018}.
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 \emph{levels}, which are assigned to experimental conditions or measurement
modalities, for example the factor \textit{`**time'} could comprise three**
`\linebreak levels: \textit{`**\SI{0}{\hour}'}, \textit{**`\SI{24}{\hour}'}, and
\textit{`**\SI{48}{\hour}'}. Such setups are attractive, because they are**
`compatible with common hypothesis tests, such as ANOVA etc.
\cite{motulskyIntuitiveBiostatisticsNonmathematical2018}: 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
\cite{krzywinskiMultidimensionalData2013}.

% TODO: Add examples of multidimensional data in biosciences
% (put the following examples into discussion? Use one example, rest into discussion?)

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 \cite{bustinReproducibilityBiomedicalResearch2014}. Here, the dependent
variable is typically the fold change
expression values derived from $\Delta\Delta$Ct calculations
\cite{brankatschkSimpleAbsoluteQuantification2012}. 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
\cite{ruedenImageJ2ImageJNext2017}. 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 \texttt{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 \cite{zhouMetascapeProvidesBiologistoriented2019}. 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
\cite{dunnExploringVisualizingMultidimensional2017}. 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
\cite{dunnExploringVisualizingMultidimensional2017}.

% \unnsubsection{Multidimensional Data in Biomedical Research}%
% \label{sec:multidimensional_data}%
% - This is the first paragraph after the sections describing cancer dissemination

% - Introduce why multidimensional data
% Modern biosciences describe processes (e.g. cell-adhesion) that are highly
% dependent on multiple experimental conditions, like time and concentration
% \cite{reblTimedependentMetabolicActivity2010, mckayCellcellAdhesionMolecules1997}
% - Go into more detail how and why cell adhesion is highly time-dependent, providing real citable sources for these claims
% - The awareness for complex datasets is rising in biosciences, as tutorials for
% multidimensional data are available
% \cite{krzywinskiMultidimensionalData2013, dunnExploringVisualizingMultidimensional2017}.

% - What is multidimensional data
% - Explain in detail, introduce terms like dependent variable, independent variable, factors

% - Describe in three examples on methods how multidimensional data is generated in biosciences
% - Provide real citable sources that show examples for every example!

% An example of multidimensional data is (multiplex) RT-qPCR, a method that
% measures the expression of multiple genes in multiple samples. The data can be
% structured with the dependent variable being the foldchange expression values
% (resulting from delta delta Ct according to
% \citet{brankatschkSimpleAbsoluteQuantification2012}), and the independent

```
% variables being the genes and the samples. The data is grouped into foldchange
% expression values as dependent variables, and

% Another hallmark example for multidimensional data are microscopy data. Stacks
% of images themselves are three-dimensional (single black-white channel),
% however, multidimensional data refers mostly to the resulting data after image
% analysis. For example, the dependent variable could a feature quantified from
% each image (e.g. number of cells). The independent factors can increase
% drastically, including experimental conditions grouped into well-plate
% coordinates (e.g. 96 wellplates), or Z-position (confocal microscopy), or time
% (time-lapse microscopy).



% Another source of multidimensional data are aggregated big-data, such as the
% result from GO-term enrichment analysis tools like \texttt{Metascape}
% \cite{zhouMetascapeProvidesBiologistoriented2019}. This Data integrates multiple
% dependent variables like foldchange expression and p-values, and the data is
% grouped into factors like gene, gene ontology terms and ontology classes from
% multiple online databases. Since these factors apply to every experimental
% conditions, GO analysis can result in highly complex and nested datasets.
% Although \texttt{Metascape} returns a graphical user interface with numerous
% summarizing graphs, the supplied Excel sheets can be non-readable for
% humans, making it difficult to extract the information that is relevant for
% specific research hypotheses.



% ============================================================================
\unnsubsection{Nontransparencies in Biomedical Data Analyses}%
\label{sec:intro_nontransparencies}%
%%%%%%%%%%%%%%%
The advent of advanced technologies in biosciences has ushered in an era of \emph{big
    data}, characterized by unprecedented volumes and complexities of
data~\cite{bubendorfHighthroughputMicroarrayTechnologies2001,
yangScalabilityValidationBig2017,
    ekmekciIntroductionProgrammingBioscientists2016}. This rise has been paralleled
by significant challenges in data analysis, particularly impacting the
reproducibility of scientific research. Studies such as the
\citet{baker500ScientistsLift2016} survey revealed that more than
\SI{70}{\percent} of researchers have tried and failed to reproduce another
scientist's experiments, highlighting a reproducibility crisis that questions
the reliability of scientific findings
\cite{begleyReproducibilityScienceImproving2015, ioannidisWhyMostPublished2005}.

\emph{Reproducibility} is considered foundational to scientific research,
ensuring that findings are reliable and verifiable. Still, its meaning requires
precise definition \cite{goodmanWhatDoesResearch2016}. The common understanding
of scientific reproduction implies not only that detailed information is
provided to enable independent repetition (\emph{transparency}), but also that
time and effort is invested into repeating the experiments
(\emph{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%
\footnotequote{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.}
{flierProblemIrreproducibleBioscience2022}
nor possible for most publications \cite{flierProblemIrreproducibleBioscience2022,
    pengReproducibleResearchComputational2011}. Hence, the meaning of
reproducibility is confined to \emph{transparency}, a concept that has
been applied to many fields, including clinical trials
\cite{goodmanWhatDoesResearch2016,
    committeeonstrategiesforresponsiblesharingofclinicaltrialdataSharingClinicalTrial2015}.

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 \SI{40}{\percent} of 127 articles, making independent
verification impossible \cite{witwerDataSubmissionQuality2013}. The same study
also found that re-analysis of data often times did not support the original
conclusions. Furthermore, \SI{44}{\percent} of 233 preclinical articles describe
statistical tests insufficiently, while few don't describe them at all
\cite{gosselinInsufficientTransparencyStatistical2021}. Another study reviewed
147 papers in the field of optometrics and found that \SI{91}{\percent} did not
discuss their rationale of correcting p-values for multiple comparisons (e.g.
Bonferroni correction) \cite{armstrongWhenUseBonferroni2014}. 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 \cite{pernegerWhatWrongBonferroni1998,
    moranArgumentsRejectingSequential2003, sullivanFactsFictionsHandling2021}. In
general, $P$-values are target of extreme scrutiny and also the cause of many
arguments, which themselves are of questionable statistical reasoning%
\footnotequote{Arguing about the P value is like focusing on a single
    misspelling, rather than on the faulty logic of a
    sentence}{leekStatisticsValuesAre2015}%
\cite{leekStatisticsValuesAre2015}. Additionally, statistical illiteracy is a
well-known problem among clinicians
\cite{lakhlifiIllusionKnowledgeStatistics2023}. Among biomedical researchers,
\SI{77}{\percent} state that they have not received formal training in data
literacy, including visualization and public deposition of data, although they
understand its high relevance \cite{federerDataLiteracyTraining2016}.
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
\cite{mesirovAccessibleReproducibleResearch2010, smithJournalOpenSource2018,gomez-
lopezPrecisionMedicineNeeds2019}.
Therefore, nontransparencies in biomedical analyses are not only caused by a
habit%
\footnotequote{[...] old habits die hard, and many will be unwilling to discard the
    hours spent learning existing systems.}{pengReproducibleResearchComputational2011}%
of insufficient reporting, but could be exacerbated by the confusions caused
by currently available methodologies and the lack of proper training.
```

\unnsubsection{Semi-Big Data: Big Enough to Cause Problems}%
\label{sec:intro_semi_big_data}%
Recent advances in big data analysis have significantly improved the standardization of both raw data availability and processing pipelines \cite{gomez-cabreroDataIntegrationEra2014}.
Particularly in RNAseq analysis, automation and the use of sophisticated software have established standards that enhance reproducibility across studies. For example, tools such as \texttt{STAR} and \texttt{HISAT} for sequence alignment, and \texttt{Cufflinks} and \texttt{DESeq} for differential expression analysis, rely on scripts that standardize processing steps to produce repeatable and verifiable results \cite{dobinSTARUltrafastUniversal2013, kimHISATFastSpliced2015,trapnellDifferentialGeneTranscript2012,loveModeratedEstimationFold2014}.
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 \emph{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 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 \cite{krzywinskiMultidimensionalData2013}.

Researchers often revert to basic tools such as \textit{Microsoft Excel} for analyzing these semi-big datasets \cite{incertiYouStillUsing2019a}. 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.

```
\unnsubsection{The Shortcomings of Common Biomedical Analysis Tools}%
\label{sec:intro_nontransparent_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 \cite{leekStatisticsValuesAre2015,
    pengReproducibleResearchComputational2011,
    mesirovAccessibleReproducibleResearch2010, localioStatisticalCodeSupport2018}.

A particularly illustrative example is \textit{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 \textit{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
\cite{gosselinInsufficientTransparencyStatistical2021,
localioStatisticalCodeSupport2018}. Furthermore, \textit{GraphPad Prism} still
requires manual data entry and lacks the robustness and automation necessary for
handling multidimensional or semi-big data. Although, \textit{GraphPad Prism} is
compatible \textit{``multiple variable tables''} \dashedsentence{similar to
`long-form tables known from \citet{wickhamTidyData2014}}, but does not
automatically graph these kinds of tables, but only user specified subsets
\cite{GraphPadPrism102024}.

Moreover, \textit{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
\cite{ziemannGeneNameErrors2016}. To compensate for these shortcomings,
\textit{Microsoft} has recently integrated a Python interpreter into
\textit{Excel}, allowing researchers to automate tasks and analyze data
efficiently and correctly \cite{microsoftexcelAnnouncingPythonExcel2023}.

Indeed, many common tools in biomedicine allow for scripting or automation to
handle semi-big data more effectively. For example, \texttt{Fiji/ImageJ}, a
popular image processing platform, supports extensive macro and scripting
capabilities \cite{ruedenImageJ2ImageJNext2017}. These features enable researchers to automate
batch processing of
image data, streamlining tasks that would otherwise require laborious manual
input. Similarly, \texttt{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 \cite{PyMOL2024, rigsbyUsingPyMOLApplication2016}.

Although automation scripts used in tools like \texttt{Fiji/ImageJ} and
```

\texttt{PyMOL} improve transparency for publishing singular data analysis pipelines, they still face challenges that can impede their reproducibility \cite{pengReproducibleResearchComputational2011, sandveTenSimpleRules2013}: 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 \cite{sandveTenSimpleRules2013}.

On the other hand, when scripts are designed to be more generalized and distributed—for instance, as a \texttt{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 \cite{narztReusabilityConceptProcess1998,wilkinsonFAIRGuidingPrinciples2016}. However, this approach also comes
with its own set of challenges \cite{sandveTenSimpleRules2013}. Often, these generalized tools often lack comprehensive user-manuals (\emph{documentation}) 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.

% ==
\unnsubsection{Modern Standards of Software Development}
\label{sec:intro_code_quality}
%%%%%%%
A main reason to write software is to define re-usable instructions for task automation~\cite{narztReusabilityConceptProcess1998}.
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~\cite{sandveTenSimpleRules2013}. Hence, modern journals aim to enforce standards to software development, including software written and used by biological researchers~\cite{smithJournalOpenSource2018}. Here, we provide a brief overview of the standards utilized by \texttt{plotastic} that to ensure its reliability and reproducibility by the scientific community~\cite{pengReproducibleResearchComputational2011}.

Modern software development is a long-term commitment of maintaining and improving code after initial release~\cite{boswellArtReadableCode2011}. Hence, it is good practice to write the software such that it is scalable, maintainable and usable. \emph{Scalability} or, to be precise, \emph{structural scalability} means that the software can easily be expanded with new features without major modifications to its architecture \cite{bondiCharacteristicsScalabilityTheir2000}. This is achieved by writing the software in a modular fashion, where each module is responsible for a single function. \emph{Maintainability} means that the software can easily be fixed from bugs and adapted to new requirements \cite{kazmanMaintainability2020}. 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~\cite{boswellArtReadableCode2011}. \emph{Usability} is hard to define~\cite{brookeSUSQuickDirty1996}, 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 \texttt{Git}, which is considered the industry standard for software development~\cite{chaconGitBook2024}. \texttt{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 \textit{``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~\cite{duvall2007continuous}. 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~\cite{10.5555/2161638}. The main type of software testing is
unit testing, which tests the smallest units of the codebase (functions and
classes) in isolation (\autoref{lst:unit_test}).

\def\mycaption{ 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 ``\texttt{test\_}'' and is placed in a
`   file that ends with the suffix ``\texttt{\_test.py}''. The test function is
`   executed by the testing framework \texttt{pytest}. Note that code after
  ``\texttt{\#}'' is considered a comment and won't be executed.}
`\begin{lstlisting}[
  language=Python,
  style=pythonstyle,
  label=lst:unit_test,
  caption=\mycaption,
  ]
# 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
\end{lstlisting}

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. The
most popular software testing framework for Python is \texttt{pytest}, which is
utilized by \texttt{plotastic}~\cite{pytestx.y}.
```

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.

# Chapter 1 abstract

Multiple myeloma involves early dissemination of malignant plasma cells across the bone marrow; however, the initial steps of dissemination remain unclear. Human bone marrow- derived mesenchymal stromal cells (hMSCs) stimulate myeloma cell expansion (e.g., IL-6) and simultaneously retain myeloma cells via chemokines (e.g., CXCL12) and adhesion factors. Hence, we hypothesized that the imbalance between cell division and retention drives dissemination. We present an \textit{in vitro} model using primary hMSCs co-cultured with INA-6 myeloma cells. Time-lapse microscopy revealed proliferation and attachment/detachment dynamics. Separation techniques (V-well adhesion assay and well plate sandwich centrifugation) were established to isolate MSC-interacting myeloma subpopulations that were characterized by RNAseq, cell viability and apoptosis. Results were correlated with gene expression data ($n=837$) and survival of myeloma patients ($n=536$). On dispersed hMSCs, INA-6 saturate hMSC-surface before proliferating into large homotypic aggregates, from which single cells detached completely. On confluent hMSCs, aggregates were replaced by strong heterotypic hMSC-INA-6 interactions, which modulated apoptosis time-dependently. Only INA-6 daughter cells (\nMAina) detached from hMSCs by cell division but sustained adherence to hMSC-adhering mother cells (\MAina). Isolated \nMAina indicated hMSC-autonomy through superior viability after IL-6 withdrawal and upregulation of proliferation-related genes. \MAina upregulated adhesion and retention factors (CXCL12), that, intriguingly, were highly expressed in myeloma samples from patients with longer overall and progression-free survival, but their expression decreased in relapsed myeloma samples. Altogether, \textit{in vitro} dissemination of INA-6 is driven by detaching daughter cells after a cycle of hMSC-(re)attachment and proliferation, involving adhesion factors that represent a bone marrow-retentive phenotype with potential clinical relevance. %

# Chapter 1 Discussion

In this study, we developed an \textit{in vitro} model to investigate the attachment/detachment dynamics of \INA\ cells to/from hMSCs and established methods to isolate the attached and detached intermediates \nMAina\ and \MAina. Secondly, we characterized a cycle of (re)attachment, division, and detachment, linking cell division to the switch that causes myeloma cells to detach from hMSC adhesion (\autoref{fig:7}). Thirdly, we identified clinically relevant genes associated with patient survival, where better or worse survival was based on the adherence status of \INA\ to hMSCs.

\INA\ cells emerged as a robust choice for studying myeloma dissemination \textit{in vitro}, showing rapid and strong adherence, as well as aggregation exceeding MSC saturation. The IL-6 dependency of \INA\ enhanced the resemblance of myeloma cell lines to patient samples, with \INA\ ranking 13th among 66 cell lines \cite{sarinEvaluatingEfficacyMultiple2020}. Despite variations in bone marrow MSCs between multiple myeloma and healthy states, we anticipated the robustness of our results, given the persistent strong adherence and growth signaling from MSCs to \INA\ during co-cultures \cite{dotterweichContactMyelomaCells2016}.

We acknowledge that \INA\ cells alone cannot fully represent the complexity of myeloma aggregation and detachment dynamics. However, the diverse adhesive properties of myeloma cell lines pose a challenge. We reasoned that attempting to capture this complexity within a single publication would not be possible. Our focus on \INA\ interactions with hMSCs allowed for a detailed exploration of the observed phenomena, such as the unique aggregation capabilities that facilitate the easy detection of detaching cells \textit{in vitro}. The validity of our data was demonstrated by matching the \textit{in vitro} findings with the gene expression and survival data of the patients (e.g., \textit{CXCL12}, \textit{DCN}, and \textit{TGM2} expression, \(n=873 \)), ensuring biological consistency and generalizability regardless of the cell line used. The protocols presented in this study offer a cost-efficient and convenient solution, making them potentially valuable for a broader study of cell interactions. We encourage optimizations to meet the varied adhesive properties of the samples, such as decreasing the number of washing steps if the adhesive strength is low. We caution against strategies that average over multiple cell lines without prior understanding their diverse attachment/detachment dynamics, such as homotypic aggregation. Such detailed insights may prove instrumental when considering the diversity of myeloma patient samples across different disease stages \cite{kawanoHomotypicCellAggregations1991, okunoVitroGrowthPattern1991}.

The intermediates, \nMAina\ and \MAina, were distinct but shared similarities in response to cell stress, intrinsic apoptosis, and regulation by p53. Unique regulatory patterns were related to central transcription factors: E2F1 for \nMAina; and NF-κB, SRF, and JUN for \MAina. This distinction may have been established through antagonism between p53 and the NF-κB subunit RELA/p65 \cite{wadgaonkarCREBbindingProteinNuclear1999, websterTranscriptionalCrossTalk1999}. Similar regulatory patterns were found in transwell experiments with RPMI1-8226 myeloma cells, where direct contact with the MSC cell line HS5 led to NF-κB signaling and soluble factors to E2F signaling \cite{dziadowiczBoneMarrowStromaInduced2022}.

```
% == Fig. 7 ====================================================
\includeimage[1][{\fmleft} 6.8in {\fmright} {\fmtop}]{
   FIGS/figures9_7.pdf
}\figcaption[fig:7]{
   Proposed model of “Detached Daughter Driven Dissemination” (DDDD) in
   aggregating multiple myeloma. \tile{Heterotypic Interaction} Malignant
   plasma cells colonize the bone marrow microenvironment by adhering to an MSC
   (or osteoblast, ECM, etc.) to maximize growth and survival through paracrine
   and adhesion mediated signaling, even if contact may trigger initial
   apoptosis. Gene expression will focus on establishing a strong anchor within
   the bone marrow, but also on attracting other myeloma cells (via secretion
   of ECM factors and CXCL12/CXCL8, respectively). \tile{Cell Division} Cell
   fission can generate one daughter cell that no longer adheres to the MSC
   (nMA). \tile{Homotypic Interaction} If myeloma cells have the capacity to
   grow as aggregates, the daughter cell stays attached to their MSC-adhering
   mother cell (MA). \tile{Re-Adhesion} The daughter cell “rolls around” the
   mother cell until it re-adheres to the MSC. Our model estimates the rolling
   duration to be \SIrange{1}{10}{\hour} long. \tile{Proliferation \& Saturation} We
   estimate that a single myeloma cell covers one MSC completely after roughly
   four population doublings. When heterotypic adhesion is saturated,
   subsequent daughter cells benefit from a homotypic interaction, since they
   stay close to growth-factor secreting MSCs and focus gene expression on
   proliferation (e.g. driven by E2F) and not adhesion (driven by NF-κB).
   \tile{Critical Size} Homotypic interaction is weaker than heterotypic
   interaction, and each cell fission destabilizes the aggregate. Hence,
   detachment of myeloma cells may depend mostly on aggregate size.
   \tile{Dissemination} After myeloma cells have detached, they gained a
   viability advantage through IL-6-independence (with unknown duration), which
   enhances their survival outside of the bone marrow and allows them to spread
   throughout the body.
}
```

The first subpopulation, \nMAina, represented proliferative and disseminative cells; They drove detachment through cell division, which was regulated by E2F, p53, and likely their crosstalk \cite{polagerP53E2fPartners2009}. \nMAina upregulate cell cycle progression genes associated with worse prognosis, because proliferation is a general risk factor for an aggressive disease course \cite{hoseProliferationCentralIndependent2011}. Additionally, \nMAina\ survived IL-6 withdrawal better than \CMina\ and \MAina, implying their ability to proliferate independently of the bone marrow \cite{bladeExtramedullaryDiseaseMultiple2022}. Indeed, xenografted \INA\ cells developed autocrine IL-6 signaling but remained IL-6-dependent after explantation \cite{burgerGp130RasMediated2001c}. The increased autonomy of \nMAina\ cells can be explained by the upregulation of \textit{IGF-1}, being the major growth factor for myeloma cell lines \cite{sprynskiRoleIGF1Major2009}. Other reports characterized disseminating cells differently: Unlike \nMAina, circulating myeloma tumor cells were reported to be non-proliferative and bone marrow retentive \cite{garcesTranscriptionalProfilingCirculating2020}. In contrast to circulating myeloma tumor cells, \nMAina\ were isolated shortly after detachment and therefore these cells are not representative of further steps of dissemination, such as intravasation, circulation or intravascular arrest \cite{zeissigTumourDisseminationMultiple2020}. Furthermore, Brandl et al. described proliferative and disseminative myeloma cells as separate entities, depending on the surface expression of CD138 or JAM-C \cite{akhmetzyanovaDynamicCD138Surface2020, brandlJunctionalAdhesionMolecule2022}. Although CD138 was not differentially regulated in \nMAina\ or \MAina, both subpopulations upregulated JAM-C, indicating disease progression \cite{brandlJunctionalAdhesionMolecule2022}.

Furthermore, \nMAina\ showed that cell division directly contributed to dissemination. This was because \INA\ daughter cells emerged from the mother cell with distance to the hMSC plane in the 2D setup. A similar mechanism was described in an intravasation model in which tumor cells disrupt the vessel endothelium through cell division and detach into blood circulation \cite{wongMitosismediatedIntravasationTissueengineered2017}. Overall, cell division offers key mechanistic insights into dissemination and metastasis.

The other subpopulation, \MAina, represented cells retained in the bone marrow; \linebreak \MAina\ strongly adhered to MSCs, showed NF-κB signaling, and upregulated several retention, adhesion, and ECM factors. The production of ECM-associated factors has recently been described in MM.1S and RPMI-8226 myeloma cells \cite{maichlIdentificationNOTCHdrivenMatrisomeassociated2023}. Another report did not identify the upregulation of such factors after direct contact with the MSC cell line HS5; hence, primary hMSCs may be crucial for studying myeloma-MSC interactions \cite{dziadowiczBoneMarrowStromaInduced2022}. Moreover, \MAina\ upregulated adhesion genes associated with prolonged patient survival and showed decreased expression in relapsed myeloma. As myeloma progression implies the independence of myeloma cells from the bone marrow \cite{bladeExtramedullaryDiseaseMultiple2022,sarinEvaluatingEfficacyMultiple2020}, we interpreted these adhesion genes as
mediators of bone marrow retention, decreasing the risk for dissemination and thereby potentially prolonging patient survival. However, the overall impact of cell adhesion and ECM on patient survival remains unclear. Several adhesion factors have been proposed as potential therapeutic targets \cite{brandlJunctionalAdhesionMolecule2022, bouzerdanAdhesionMoleculesMultiplea}. Recent studies
have described the prognostic value of multiple ECM genes, such as those driven by NOTCH \cite{maichlIdentificationNOTCHdrivenMatrisomeassociated2023}. Another study focused
on ECM gene families, of which only six of the 26 genes overlapped with our gene set (\apdxref{apdx:supplemental}{tab:S2}) \cite{eversPrognosticValueExtracellular2023}. The expression of only one gene (\textit{COL4A1}) showed a different association with overall survival than that in our cohort. The lack of overlap and differences can be explained by dissimilar definitions of gene sets (homology \textit{vs.} gene ontology), methodological discrepancies, and cohort composition.

In summary, our \textit{in vitro} model provides a starting point for understanding the initiation of dissemination and its implications for patient survival, providing innovative methods, mechanistic insights into attachment/detachment, and a set of clinically relevant genes that play a role in bone marrow retention. These results and methods might prove useful when facing the heterogeneity of disseminative behaviors among myeloma cell lines and primary materials.

Chapter 2 abstract

> \texttt{plotastic} addresses the challenges of transitioning from exploratory data analysis to hypothesis testing in Python's data science ecosystem. Bridging the gap between \texttt{seaborn} and \texttt{pingouin}, this library offers a unified environment for plotting and statistical analysis. It simplifies the workflow with user-friendly syntax and seamless integration with familiar \texttt{seaborn} parameters (y, x, hue, row, col). Inspired by \texttt{seaborn}'s consistency, \texttt{plotastic} utilizes a \texttt{DataAnalysis} object to intelligently pass parameters to \texttt{pingouin} statistical functions. Hence, statistics and plotting are performed on the same set of parameters, so that the strength of \texttt{seaborn} in visualizing multidimensional data is extended onto statistical analysis. In essence, \texttt{plotastic} translates \texttt{seaborn} parameters into statistical terms, configures statistical protocols based on intuitive plotting syntax and returns a \texttt{matplotlib} figure with known customization options and more. This approach streamlines data analysis, allowing researchers to focus on correct statistical testing and less about specific syntax and implementations.

# Chapter 2 Discussion

As awareness of the complexities associated with multidimensional data in biomedical research increases, there is a growing demand for tools that not only simplify analysis but also enhance its intuitiveness and effectiveness \cite{dunnExploringVisualizingMultidimensional2017}. \texttt{plotastic} is designed to meet this demand by seamlessly integrating data visualization with inferential statistics, making sophisticated statistical methods accessible to researchers of all expertise levels. This integration could be pivotal as it allows the visualization of data —how it is grouped and presented— to directly guide the statistical analysis, reducing the need for in-depth statistical knowledge and ensuring that the analyses are intuitively aligned with the visual aspects of the data. This approach could not only simplify the analytical process but also enhance the transparency and reproducibility of research findings.

\textbf{Statistical Features:}
A detailed list of implemented and planned features is provided on the GitHub page of the project \cite{kuricMarkur4Plotastic2024}. \texttt{plotastic} is comprehensive in its current scope, incorporating a robust suite of statistical tests that cater to a wide range of research needs. It includes assumption tests for normality, homoscedasticity, and sphericity, alongside classical statistical tests such as ANOVA and t-tests, available in both parametric and non-parametric forms, as well as paired and unpaired variants. However, its reliance on the \texttt{pingouin} library means that \texttt{plotastic} is subject to the same limitations as \texttt{pingouin} itself. For instance, it does not yet support survival analysis tools like log-rank tests and Kaplan-Meier plots, which are critical for certain biomedical applications. While there are external packages that offer these capabilities, integrating them into \texttt{plotastic} could significantly expand its utility and provide a more unified user experience \cite{davidson-pilonLifelinesSurvivalAnalysis2019}.

One known issue in \texttt{plotastic} is its handling of multiple testing corrections. Currently, \texttt{plotastic} might not correctly apply these corrections when the data is split across different facets with their own y-axes (facetted by \texttt{row} and \texttt{col} keywords), which can lead to potentially incorrect statistical inferences. This is a fixable issue, and plans are in place to address it in upcoming versions to ensure that corrections for multiple testing are appropriately applied across the complete dataset. Additionally, bivariate analysis tools like correlation and regression are not yet implemented, since \texttt{plotastic} focused on data with a categorical x-axis, which is more common in biomedical research.

\textbf{Plotting Features:} The plotting capabilities of \texttt{plotastic} employ all of \texttt{seaborn}'s non-facetgrid plotting functions (e.g. \texttt{sns.boxplot()}), which include a wide range of plot types but may not cover all possible visualizations \cite{waskomSeabornStatisticalData2021}. Future versions could expand the range of specialized plots, for example QQ-plots. \texttt{plotastic} focuses on offering both high- and low-level plotting configuration: \texttt{Multiplots} automate overlaying multiple plot types, which is extremely useful for displaying raw data points alongside aggregated statistics (barplots, boxplots, etc.), a feature that can be cumbersome to implement manually. Low-level plotting configuration is supported just like in \texttt{seaborn}, since plotastic uses \texttt{matplotlib} as its backend. This level of flexibility is unique to \texttt{plotastic}, serving both beginners and advanced users.

\textbf{Plot Annotation:} Annotating statistical results into plots (e.g. ***

above barplots) is a key requirement in modern biomedical journals and could be key feature why researchers choose proprietary software like \textit{GraphPad Prism} over other solutions. \texttt{plotastic} automates this process as well, making it a strong competitor to other statistical software. This is especially useful for re-arranging plots, since the statistical annotations are automatically updated when the plot is re-drawn. This feature is unique to \texttt{plotastic} and could be a key selling point for the software.

\textbf{Software Testing:} The development of \texttt{plotastic} adheres to modern software engineering principles to ensure reliability and maintainability. The project utilizes continuous integration practices, which means that with every change to the codebase, a comprehensive test suite is automatically run to identify potential bugs and ensure that new contributions do not disrupt existing functionalities. This test suite covers approximately \SI{79}{\percent} of the testable lines of code, a statistic tracked automatically by an independent service called \texttt{codecov}, highlighting a strong commitment to software quality \cite{Codecov2024}.

\textbf{Documentation:}
Documentation serves as a critical resource for enhancing user experience and adoption, especially for software aimed at users with varying levels of expertise. Currently, \texttt{plotastic}'s documentation is focused on basic functionalities. These include detailed installation instructions, example analyses using five test datasets from \texttt{seaborn} that are commonly used in teaching statistics, guidelines on dimension switching with commands like \texttt{DataAnalysis.switch()}, and tutorials on constructing and configuring plots, annotating statistical data, and utilizing multiplot capabilities.

However, the documentation of \texttt{plotastic} could be significantly enhanced. Currently, it lacks a dedicated website, relying instead on GitHub-hosted Jupyter notebooks. While useful, these notebooks are not the most user-friendly or maintainable format for documentation as they can be challenging to navigate and don't update synchronously with software changes. A more robust approach would involve leveraging services like \texttt{Read the Docs} or \texttt{Sphinx} to generate and host documentation directly from the codebase \cite{ReadDocs2024, Sphinx2024}. This would not only ensure
that the documentation remains up-to-date with the latest software developments but also provide a more accessible and navigable user experience, meeting the expectations of users who prefer a dedicated website for software documentation.

\textbf{Usability for Non-Statisticians:} \texttt{plotastic} aims to make statistical analysis more accessible to researchers without extensive statistical training by intuitively mapping plotting concepts to statistical operations. To the author's knowledge, this approach is unique to \texttt{plotastic} and has great potential to make statistics easier and educational for non-statisticians. Still, the software requires responsible and self-critical usage, as emphasized by the thorough disclaimer on its GitHub page regarding the software's statistical robustness, \cite{kuricMarkur4Plotastic2024}. The disclaimer highlights that while \texttt{plotastic} can facilitate gaining practical experience with statistics and provide a preliminary analysis, it is not a substitute for professional statistical consultation. It is designed to aid users in generating publication-grade figures and performing statistical tests, provided they have a basic understanding of the procedures involved or have their results verified by a statistician. To enhance usability for non-statisticians, \texttt{plotastic} could incorporate a system to suggest appropriate statistical tests based on data characteristics, like parametric tests for normally distributed data. This feature would guide users in selecting the correct tests, thereby augmenting the tool's functionality and broadening its appeal. Additionally, the GitHub page provides critical guidelines for responsible statistical practice, urging users to document their work in detail, understand the limitations of the tests applied, and consult professionals to validate their findings, ensuring that \texttt{plotastic} supports but does not replace thorough statistical analysis \cite{sandveTenSimpleRules2013, kuricMarkur4Plotastic2024}.

\textbf{Usability for Non-Programmers:}
Despite the advantages of \texttt{plotastic}, its adoption among non-programmers in biomedicine may be challenging due to its reliance on a command-line interface (CLI), which is less intuitive for those accustomed to graphical user interfaces (GUIs). However, the integration of advanced artificial intelligence technologies, such as ChatGPT, presents a compelling case for embracing CLI. Indeed, ChatGPT is believed to potentially revolutionize medical research \cite{ruksakulpiwatUsingChatGPTMedical2023}.

Unlike GUIs, CLIs are highly compatible with text-based AI technologies, which can significantly lower the barrier to entry. In fact, both ChatGPT-3.5 and -4 demonstrate impressive performance in python \cite{arefinUnmaskingGiantComprehensive2023}. This is a game changer, since researchers can now use similair tools as programmers and are only limited by their methodological expertise to formulate a correct prompt \cite{qureshiAreChatGPTLarge2023}\footnotequote{You can now recognize and learn the language of almost anything with structure, and you can translate it to anything with structure — so text-protein, protein-text. [...] Everybody is a programmer, and the programming language of the future is called
`human.'}{kelleherNVIDIACEOThis2024}. For instance, when a software is not
`working as intended, users of a GUI are likely to be stuck without help or further research. Users of a CLI however, could utilize ChatGPT to ask for code-corrections or explanations of the code line-by-line, but also for advice on how to proceed with a statistical analysis and how to implement new features (e.g. editing a plot). Attempts to integrate AI into GUIs however have proven challenging \cite{gaoASSISTGUITaskOrientedDesktop2024}.

Still, ChatGPT requires responsible use, as it is not sufficient as a standalone tool for statistical analysis \cite{ordakChatGPTSkillsStatistical2023}. It should also be noted that \texttt{plotastic} is not yet known to ChatGPT, but could be included in future versions, depending on the popularity of \texttt{plotastic}.

Overall, the transition to a new data analysis software, especially one that incorporates coding, presents a learning curve. However, the advantages of plotastic in terms of analytical clarity, speed, and depth are anticipated to outweigh these initial challenges.

\textbf{Adoption and Open-Source Contributions:}
The adoption rate of \texttt{plotastic} is a critical factor for its sustainability, particularly in the open-source environment where community contributions can significantly support the author in improving and maintaining the software. Since its publication in the Journal of Open Source Software on March 9, 2024, \texttt{plotastic} has garnered attention with 41 visits and 8 \emph{stars} (similar to a `like' on social media platforms) on its GitHub page. `This level of engagement, while modest, shows initial interest and potential for growth. Active involvement from the community is essential for ongoing improvements; hence, efforts are being made to enhance the software's documentation and structure to attract more contributors: \texttt{plotastic's} GitHub page shows a detailed outline of the software's architecture as a class diagram in unified modeling language (UML) format, helping potential contributors orient themselves within \texttt{plotastic}'s several modules and classes (shown in \apdxref{subapdx:classdiagr}). But further efforts are required, e.g. only few functions are documented with docstrings, which help understanding the purpose and usage of each function. Still, \texttt{plotastic} is a general-purpose data analysis software designed not only for biologists but for a broad range of scientific disciplines, making it a versatile tool with promising potential for wider adoption.

\textbf{Contributions to Methodological Transparency and Biomedicine:}
\texttt{plotastic} standardizes statistical analysis by ensuring that it is performed alongside visual representations. This integrated approach simplifies both analysis and interpretation, facilitating smooth replication of analyses. Although it streamlines the data analysis process, it is not a complete solution to the reproducibility crisis in scientific research. Researchers must still possess a basic understanding of data analysis principles and be cautious about their reliance on scripting solutions like Python, which is less familiar to some biomedical researchers.

Statistical literacy and lack of training is a well-documented challenge among clinicians and biomedical researchers, decreasing the confidence in presenting their analyses in detail \cite{lakhlifiIllusionKnowledgeStatistics2023, federerDataLiteracyTraining2016}. Since \texttt{plotastic} alleviates some need for statistical knowledge by automating the configuration of statistical tests, the room for error is reduced, and the user can lay off some responsibility to the software, gaining confidence in presenting their analysis transparently.

Furthermore, \texttt{plotastic}'s compatibility with the Jupyter ecosystem leverages \textit{``simple, intuitive ways to both capture and embed computational work `directly into our papers''} as advocated by \citet{mesirovAccessibleReproducibleResearch2010}. This integration makes \texttt{plotastic} not just a tool for analysis but also a means of enhancing the accessibility and reproducibility of scientific work. As \citet{pengReproducibleResearchComputational2011} suggests, the exploration of data and analysis code may often be sufficient to verify the quality of scientific claims. This seems plausible, given that statistical tests themselves pose rigorous requirements on the data, and the results are often not interpretable without the context data. Combining \texttt{plotastic} with Jupyter Notebooks provides a compelling solution to transparently integrate and document both intermediate results and analytical processes, thus furthering scientific rigor and replicability.

Overall, \texttt{plotastic} is useful statistical tool with the potential to improve methodological transparency and reproducibility of research in biomedicine.

# Summarising Discussion

\unnsubsection{Semi-Automation was Critical for Establishing \textit{in vitro} Methods}%
\label{sec:discussion_semi_automated_analysis}%
\textit{In vitro} research is valued for their speed at creating precise data \cite{moleiroCriticalAnalysisAvailable2017}. In this work, the development and publication of innovative \textit{in vitro} methodologies necessitated the adoption of semi-automated data analysis tools. These novel methods introduced complexities that span multiple experimental parameters, making the results multidimensional. This demanded precise, efficient and standardized data handling capabilities which were facilitated by Python tools like \texttt{seaborn} and \texttt{plotastic} \cite{waskomSeabornStatisticalData2021,kuricPlotasticBridgingPlotting2024}.

\textbf{Inherent Multidimensionality of Adhesion Studies:}
Cell adhesion studies often involve multiple independent parameters, posing significant analytical challenges. Two critical dimensions are particularly notable: \emph{`Subpopulation'} and \emph{`Time'}. Analyzing cell adhesion often involves isolation of adherent and non-adherent subpopulations, effectively introducing \textit{`Subpopulation'} as a vital dimension in the dataset `\cite{dziadowiczBoneMarrowStromaInduced2022}. This study specifically categorized cells into three levels of MSC-interaction: \CMina, \nMAina, and \MAina. Furthermore, the dynamic nature of cell adhesion processes is profoundly influenced by the factor \emph{`Time'}, making it a crucial experimental

`parameter for investigation \cite{reblTimedependentMetabolicActivity2010, mckayCellcellAdhesionMolecules1997, bolado-carrancioPeriodicPropagatingWaves2020a}. This work includes extensive time-lapse microscopy experiments utilizing a high time resolution (\SI{1}{frame} every \SI{15}{\minute}), similar to those time resolutions used by \citet{purschkePhototoxicityHoechst333422010}. This precision was required for key mechanistic insights on hMSC-\INA interaction dynamics. These included identifying rolling movements of \nMAina daughter cells around \MAina mother cells, measuring the minimum time for \INA detachments to begin, and measuring the time required for daughter cells to re-attach to the hMSC monolayer, etc. Next to mechanistic insights, adhesion time played a crucial methodological role in this study as well: During the V-Well adhesion experiments, we did not know initially how long \INA cells required to form strong adhesion with hMSCs before pelleting \nMAina, but required a timepoint with hour precision to capture detachments after cell division that was accelerated through prior cell cycle synchronization at M-Phase.

The extensive facetting features of \texttt{seaborn} and \texttt{plotastic} were essential for visualizing these multidimensional datasets, allowing for quick exploration of the data \cite{waskomSeabornStatisticalData2021}.

\textbf{Further Contributions and Remedies to Multidimensional Complexity:} In addition to \textit{`**Subpopulation'} and \textit{**`Time'}, this study faced additional layers of complexity that were managed through semi-automated analysis.

Experiments typically involved at least three biological replicates and corresponding technical replicates. Although replicates were not treated as independent variables \dashedsentence{instead serving for displaying variance} they can add substantially to the data management workload. In this work, semi-automation nullified the manual burdens of handling replicates: \texttt{pandas} was used to automate aggregation of technical replicates into means after removing technical outliers thorugh z-score thresholding, while the jupyter notebook format allowed for reviewing filtered data for specific data losses. The removal of technical noise was especially relevant for qPCR data, where low gene expression can lead to sudden increase in Ct value (non-detects). In fact, the decision to remove or impute non-detects is under active discussion, however, available algorithms are hard to understand for non bioinformaticians, but also do not separate biological from technical variance, which is considered bad practice by \citet{motulskyIntuitiveBiostatisticsNonmathematical2018} \cite{mccallNondetectsQPCRData2014, sherinaMultipleImputationDirect2020}. Semi-automation also nullified the burden of handling biological replicates: The automatic aggregation of datapoints during plotting is a key feature of \texttt{seaborn}, on which \texttt{plotastic} was built. Without such automation, calculating means and standard deviations for simple barplots would have required extensive manual computation in \textit{Microsoft Excel}, or tedious plot configurations in \textit{Graphpad Prism} due to limited facetting functionality of multiple variable tables \cite{GraphPadPrism102024}.

```
Replicates can expand datasets as this factor comprises a lot of levels.
Similarly, the factor \textit{`Gene'} multiplied the dataset by a total of 30
`genes when validating RNAseq data with RT-qPCR. With three subpopulations, one
timepoint, eleven biological replicates, and three technical replicates, the
qPCR dataset used in this study grew to 2970 datapoints to be statistically
analyzed and visualized. This is a manageable size for manual analysis, but the
effort involved illustrates the definition of semi-big data.

Methodological variability also introduced additional dimensions: The Well Plate
Sandwich Centrifugation (WPSC) used two different techniques to dissociate
\MAina cells from the hMSC monolayer: Strong pipetting
(\emph{`Wash'}) and repeated Accutase treatment followed by magnetic activated
`cell sorting (\emph{`MACS'}). These variations, recorded as the factor
`\textit{`method'}, further complicated the dataset. Although this distinction is
`not discussed in this work \dashedsentence{rather pooled into one group}, this
showcases how protocol changes can add dimensions to the dataset that are
not necessarily relevant for the biological question but essential for
method optimizations and validation.



\textbf{Agility During Establishment of V-Well Assay:}
The concept of agility in laboratory research, inspired by the Agile Manifesto's
principle of \emph{``Responding to change over following a plan''}
`\cite{ManifestoAgileSoftware2001}, is increasingly relevant in biomedical
research \cite{westReinventingResearchAgile2018,
    quanbeckApplyingConceptsRapid2022}. This adaptive approach was particularly
crucial during the development of the V-Well adhesion assay in this study.
Semi-automation using python significantly enhanced this agility, allowing rapid
statistical testing and visualization of data, which would have taken
considerably longer if done manually. This capability enabled real-time
adjustments to the experimental technique during live microscopy sessions,
integrating raw data tables directly into Python scripts for immediate analysis.
Such an agile and adaptive work environment, facilitated by python tools and
\texttt{seaborn}, proved invaluable for refining the \textit{in vitro} methods
being developed. Additionally, the simplicity offered by \texttt{seaborn} for
complex data plotting, such as the cell cycle profiling shown in
\apdxref{subapdx:figs}{fig:S3}, which required minimal code to produce a
detailed series of 24 histograms, underscores the utility of semi-automation in
enhancing laboratory efficiency. While this work does not quantify the full
benefits of semi-automation, the author's experiences suggest significant
potential impacts on the speed and adaptability of method development in
biomedical research.




% ==========================================================================
\unnsubsection{\texttt{plotastic} Exceled in Re-Doing Statistical Analyses and Plots}%
```

\label{sec:discussion_plotastic}%
Establishing new methods of \textit{in vitro} dissemination required not just innovative experimental protocols, but also adaptive ways to visually present complex data. This need for adaptability is crucial during the publication process, where researchers must often experiment with different ways to visually represent their findings to best convey their significance. This process typically involves frequent adjustments to how data is displayed in plots. Such adjustments become especially cumbersome when subsequent adjustments are involved. Traditional tools (\textit{Microsoft Excel} or \textit{Graphpad Prism}) fail at handling semi-big data, while Python packages like \textit{seaborn} reach their limits in terms of adaptability, making the development of \texttt{plotastic} a critical step in this work.

\texttt{plotastic} addresses these challenges by not only automating statistics, but also by enhancing the adaptability of data visualization as well, making it easier to modify how data is presented without repetitive manual adjustments. The author saw four key steps that required streamlining through \texttt{plotastic}:
\begin{enumerate}
    \item {Re-arranging facets}
    \item {Plotting multiple layers of different plot types}
    \item {Statistical Re-Analysis and Re-Annotation}
    \item {Fine-Tuning for publication grade quality}
\end{enumerate}

These adjustment steps made re-plotting tedious, since a change in prior steps required a complete re-work of following steps, something which \texttt{plotastic} prevented. Its key design feature is the centralized storage of facetting parameters (\facetparams). These parameters define which data points are shown on the x-axis, what categories are highlighted by color (\texttt{hue}), and how data is grouped into separate plots (by columns and/or rows) into separate plots. This centralization means that once these parameters are set, they not only automate statistical analysis, but also can be automatically applied across all subsequent adjustments made to the plot. This contrasts with \texttt{seaborn}, where changing these parameters required adjusting multiple lines of subsequent code.

\textbf{Re-arranging Facets:} \texttt{plotastic}'s \texttt{.switch()} method allowed for easily shifting the arrangement of plots \dashedsentence{for example, switching the data represented on the x-axis with that represented by color} to explore different perspectives of the data quickly. This proved particularly useful when trying to find the most effective way to illustrate complex interactions or trends that might not be immediately apparent. In \texttt{seaborn}, changing facets is easy and proved useful during intermediate data analysis, but unfeasable when plots involved multiple layers, sophisticated style edits or statistical annotations, as this can require re-writing subsequent adjustments.

\textbf{Plotting Multiple Layers of Different Plot Types:}
Modern journal standards increasingly demand the representation of individual datapoints alongside aggregated data, for example plotting datapoints above a

bar- or boxplots. \texttt{seaborn} does not automate this, but can require calling two plotting functions in sequence, e.g. \texttt{sns.boxplot()} followed by \texttt{sns.swarmplot()}. This can be can get repetitive, as adjusting the style of these plots to match each other, e.g. defining the point size or transparency of individual data points to fit into a barplot. \texttt{plotastic} was designed for multi-layered plotting, offering single-line functions for plot combinations with pre-configured style-parameters.

\textbf{Statistical Re-Analysis and Re-Annotation}
To the author's knowledge, \texttt{plotastic}'s capability of streamlining statistical re-analysis is unique and unmatched. \texttt{seaborn} alone can not perform this without multiple lines of \texttt{statannotations} \cite{charlierTrevismdStatannotationsV02022}. \texttt{plotastic} automates the inclusion of statistical annotations directly into plots. This is a significant enhancement because it ensures that any statistical significance noted in the data is immediately visible and correctly updated whenever the data presentation is changed. This feature proved particularly useful during the peer review process of \citet{kuricModelingMyelomaDissemination2024}, where a reviewer asked for a complete statistical analysis of Chapter\,1\,\ref{fig:5}\,D, which at that time included only paired t-tests between selected groups.

\textbf{Fine-Tuning for Publication Grade Quality:}
\texttt{plotastic} simplified the creation of publication-quality figures by automating style adjustments that are typically manually coded with \texttt{matplotlib} when using \texttt{seaborn}. These include adjustments like angled x-axis labels or consistent visual styles across multiple figures, which are important for maintaining the professional appearance of published data.

\textbf{Outlook: Could \texttt{plotastic} Address a Re-Analysis Bottleneck?}
Re-doing analyses and plots is often overlooked bottlenecks in the reproducibility of scientific research, although it does overlap with two principles of the FAIR-guidelines for scientific data management and stewardship: Interoperability\footnotequote{Interoperability\,—\,the ability of data or tools from non-cooperating resources to integrate or work together with minimal effort.}{wilkinsonFAIRGuidingPrinciples2016} and Re-Usability \cite{wilkinsonFAIRGuidingPrinciples2016}. This challenge was exemplified during this work's experiments using RT-qPCR. The field of qPCR, where reproducibility issues have been notoriously prevalent. As \citet{bustinReproducibilityBiomedicalResearch2014} noted, many publications using PCR-based methods have been seriously flawed, underscoring the need for updated approaches \cite{bustinNeedTransparencyGood2013, ruiz-villalbaUseMisuseCq2021}. Furthermore, the evolution of the $\Delta\Delta$Ct formula over recent years highlights the dynamic nature of data analysis standards in biomedicine \cite{pfafflNewMathematicalModel2001a, ramakersAssumptionfreeAnalysisQuantitative2003, ruijterEfficiencyCorrectionRequired2021}. Despite these challenges, current data analysis infrastructures seldom facilitate a smooth revision or complete redoing of figures, which could hamper efforts to re-analyse and apply the latest techniques to existing datasets, which could be requested e.g. during peer-review \cite{wilkinsonFAIRGuidingPrinciples2016}. In response,

\texttt{plotastic} was specifically designed to streamline the reconfiguration and reanalysis of data visualizations. This work serves as a case study showing that \dashedsentence{according to the author's experiences} the manual effort involved was effectively reduced, making the task of re-analysis seem a lot more inviting, especially for handling semi-big data.

% ==
\unnsubsection{Conclusion 2: Demonstrating the Advantages of Semi-Automation in Biomedical Research Methodologies}%
\label{sec:discussion_conclusion_semi_automation}%
This thesis illustrates the challenges and solutions associated with managing the inherent complexity of adhesion studies and related methodologies, such as Cell Cycle profiling. These methodologies necessitate sophisticated data handling tools to address two primary challenges: (1) the multidimensionality of semi-big data and (2) the rapid iterative loop of results evaluation and protocol adjustments, a process for which \textit{in vitro} methods are valued.

\texttt{seaborn} and \texttt{plotastic} have been instrumental in addressing these challenges. \texttt{seaborn} facilitated the rapid processing of intermediate results during method development, while \texttt{plotastic} was crucial for crafting publication-grade analyses and figures, filling in the capabilities that \texttt{seaborn} lacks. This includes facilitating the easy (re-)design of visualizations and statistical analyses, which are critical for late-stage data processing.

Though this work does not provide empirical evidence quantifying the benefits of semi-automation, it serves as a practical case study demonstrating the transformative potential of such technologies in biomedical research. The integration of semi-automation tools streamlined complex \textit{in vitro} methodologies, significantly enhancing operational agility. This case study bridges biomedical research with bioinformatics, highlighting how semi-automation can reduce data analysis workloads and enable researchers to focus more on exploratory research within the laboratory setting.

To the author's experience, the gained efficiencies not only saved valuable time but also enhanced the clarity and communicative power of the research findings. This is particularly crucial in fields like myeloma dissemination, where precise and transparent data presentation is essential for advancing understanding and treatment strategies. This conclusion suggests a need for further empirical research to validate these benefits more broadly and encourage wider adoption of semi-automation tools in biomedical research.

However, adopting \texttt{plotastic} poses its own set of challenges, particularly in the realm of biomedicine where researchers may prefer graphical user interfaces (GUIs) over command-line interfaces (CLIs). While \texttt{plotastic} offers a powerful CLI that is efficient and capable of handling complex data manipulation and visualization tasks, the transition from

GUIs to CLIs can be intimidating for those accustomed to more visual interaction with software. This barrier can be mitigated by the integration of tools like ChatGPT, which can facilitate the use of CLIs by offering context understanding, code assistance, and error identification.
---

Based on the detailed inputs and the sections you provided, here's a concise one-page summary of your PhD thesis that maintains your LaTeX formatting style and encompasses the overarching themes and findings:

latex

```latex
\documentclass{article}
\usepackage{amsmath,amsfonts,amssymb,siunitx}
\usepackage{graphicx}
\usepackage{lipsum} % for generating text filler

\begin{document}
\title{Summary of PhD Thesis: Development and Semi-Automated Analysis of an \textit{in
vitro} Dissemination Model for Myeloma Cells Interacting with Mesenchymal Stromal Cells}
\author{[Your Name Here]}
\date{[Date Here]}
\maketitle

\section*{Summary}
This PhD thesis develops an innovative \textit{in vitro} model to understand the
mechanisms behind the dissemination of multiple myeloma cells in interaction with human
mesenchymal stromal cells (hMSCs). The thesis is structured into two primary chapters,
integrating biomedicine with advanced data analysis tools to bridge significant gaps in
both fields.

\textbf{Chapter 1} describes the establishment of an \textit{in vitro} model using primary
hMSCs co-cultured with INA-6 myeloma cells. Time-lapse microscopy coupled with novel
separation techniques allowed for the detailed observation and analysis of myeloma cell
proliferation, attachment, and detachment dynamics. This chapter identifies key processes
in myeloma cell dissemination, notably how myeloma cells proliferate on and detach from
hMSCs, facilitated by the differential expression of adhesion and retention factors, which
correlate with patient survival rates. These findings provide a deeper understanding of
the cellular interactions at play in myeloma dissemination and establish a foundation for
potential therapeutic targets.

\textbf{Chapter 2} introduces \texttt{plotastic}, a Python software environment designed
to ease the analysis of large, multidimensional datasets commonly generated in biomedical
research. \texttt{plotastic} enhances the reproducibility and transparency of data
analysis by seamlessly integrating advanced visualization and statistical analysis
capabilities. The software is tailored to simplify the user experience while providing
robust statistical tools, thereby addressing common issues in biomedical data analysis
such as handling semi-big data and ensuring methodological transparency.

The \textbf{summarising discussion} bridges the insights from both chapters, highlighting
the role of semi-automated data analysis in refining \textit{in vitro} methodologies. It
underscores the challenges posed by multidimensional data and the solutions provided by
\texttt{plotastic}, which significantly streamline the visualization and statistical
analysis processes. This integration not only facilitates a more efficient research
workflow but also enhances the clarity and reliability of the findings, which are crucial
for advancing biomedical research and therapeutic development.

Overall, this thesis not only advances our understanding of myeloma cell biology and
interactions but also contributes significantly to the field of data science in
biomedicine by developing and implementing a tool that enhances the accessibility and
reproducibility of complex data analyses.
```

```
\end{document}
```

This summary should fit well within a one-page limit, depending on your formatting requirements (e.g., margins, font size). Adjustments can be made to ensure it aligns perfectly with the rest of your thesis in terms of style and detail.

It's a good start, but I want a different structure that requires re-writing it completely. Instead of structuring the summary chapter per chapter, write one cohesive narrative, so that it's that both chapters are well connected. Also, please implement the word "case study" when including information from chapter 2. Also, you do not have to include a preamble, just provide the summary in latex code.

latex

```
\section*{Summary}
This PhD thesis presents a cohesive integration of biomedical research and data analysis,
focusing on the development of an \textit{in vitro} model for studying myeloma cell
dissemination and the creation of a software tool for semi-automated analysis of
multidimensional datasets. The work addresses critical challenges in understanding the
early steps of myeloma dissemination and improving the reproducibility and transparency of
data analysis in biomedical research.

The research begins with the establishment of an \textit{in vitro} model using primary
human mesenchymal stromal cells (hMSCs) co-cultured with INA-6 myeloma cells. Through
time-lapse microscopy, we observed the dynamics of myeloma cell proliferation, attachment,
and detachment. Key insights were gained into how myeloma cells interact with hMSCs,
revealing that detachment often follows cell division, driven by daughter cells emerging
from mother cells adhering to hMSCs. This process was characterized by the differential
expression of adhesion and retention factors, correlating with clinical data on patient
survival. The model provides a detailed understanding of the mechanisms behind myeloma
cell dissemination and highlights potential therapeutic targets.

Addressing the complex nature of the data generated from these experiments required
advanced analysis tools. Here, the development of \texttt{plotastic}, a Python-based
software, plays a pivotal role. \texttt{plotastic} was created to streamline the analysis
and visualization of large, multidimensional datasets, bridging the gap between
exploratory data analysis and hypothesis testing. This tool integrates seamlessly with
existing Python libraries such as \texttt{seaborn} and \texttt{pingouin}, enabling
researchers to perform comprehensive statistical analyses directly from their data
visualization setups. The software's ability to automate and standardize these processes
not only enhances efficiency but also significantly improves the reproducibility of
research findings.

The case study involving \texttt{plotastic} demonstrates its practical application in re-
analyzing and visualizing complex datasets derived from the \textit{in vitro} myeloma
dissemination model. By automating repetitive tasks and ensuring consistent statistical
analyses, \texttt{plotastic} allowed for rapid adjustments and refinements in data
presentation. This capability was crucial during the publication process, where visual
clarity and methodological transparency are paramount.

Overall, this thesis illustrates the symbiotic relationship between innovative
experimental methodologies and advanced data analysis tools. The \textit{in vitro} model
developed provides a robust platform for studying myeloma cell dissemination, while
\texttt{plotastic} addresses the critical need for reliable and reproducible data analysis
in biomedical research. Together, they offer a comprehensive approach to tackling the
challenges of multidimensional data, contributing to both the understanding of cancer
```

```
biology and the enhancement of research practices.

The findings underscore the importance of semi-automation in managing the complexities of
modern biomedical data. By integrating these methodologies, the thesis not only advances
the field of myeloma research but also sets a precedent for future studies aiming to
bridge experimental biology with data science. This approach ensures that intricate
biological phenomena can be analyzed with greater precision and transparency, ultimately
supporting the advancement of scientific knowledge and the development of effective
treatments.
```

It's good! However it is three paragraphs too long. Please shorten it, without losing information. Make every word count.

latex

```
\section*{Summary}
This PhD thesis integrates biomedical research and data analysis, focusing on an
\textit{in vitro} model for myeloma cell dissemination and a semi-automated data analysis
tool. The work addresses critical challenges in understanding early myeloma dissemination
and improving data analysis reproducibility.

Using primary human mesenchymal stromal cells (hMSCs) co-cultured with INA-6 myeloma
cells, the research explores myeloma cell proliferation, attachment, and detachment
through time-lapse microscopy. Key findings reveal that detachment often follows cell
division, driven by daughter cells. Differential expression of adhesion and retention
factors correlates with patient survival, providing insights into myeloma dissemination
mechanisms and potential therapeutic targets.

To manage complex data, the thesis develops \texttt{plotastic}, a Python-based software
that streamlines analysis and visualization of multidimensional datasets. Integrating with
\texttt{seaborn} and \texttt{pingouin}, \texttt{plotastic} automates statistical analyses,
enhancing efficiency and reproducibility. The software's application is demonstrated in a
case study re-analyzing data from the \textit{in vitro} model, facilitating rapid
adjustments and refinements in data presentation.

This thesis illustrates the synergy between experimental methodologies and advanced data
analysis tools. The \textit{in vitro} model provides a robust platform for studying
myeloma dissemination, while \texttt{plotastic} addresses the need for reliable data
analysis. Together, they offer a comprehensive approach to handling multidimensional data,
advancing both cancer biology and research practices. This integration supports precise
and transparent analysis of biological phenomena, promoting scientific progress and
effective treatments.
```

Ok, using your input, I have drafted this summary. Do you like it?

> This PhD thesis presents a cohesive integration of biomedical research and data science, focusing on the development of an \textit{in vitro} model for studying myeloma cell dissemination and the creation of a software tool for semi-automated analysis of multidimensional datasets. The work addresses critical challenges in biomedical research, including (1) understanding the

early steps of myeloma dissemination and (2) improving the efficiency of data analysis, which \dashedsentence{shown in this thesis} represents a reproducibility bottleneck for biomedical research.

Using primary human mesenchymal stromal cells (hMSCs) co-cultured with INA-6 myeloma cells, the research explores myeloma cell proliferation, attachment, and detachment through time-lapse microscopy. Key findings reveal that detachment often follows cell division, driven by daughter cells. Innovative separation techniques were developed to isolate myeloma subpopulations interacting with hMSCs, which were characterized by RNAseq, cell viability, and apoptosis. Differential expression of adhesion and retention factors correlates with patient survival, providing insights into myeloma dissemination mechanisms and potential therapeutic targets.

To manage complex data, develops \texttt{plotastic}, a Python-based software that streamlines analysis and visualization of multidimensional datasets. Integrating with \texttt{seaborn} and \texttt{pingouin}, \texttt{plotastic} semi-automates statistical analyses, enhancing efficiency and reproducibility. The thesis reflects on the experiences gathered from applying \texttt{plotastic} to the \textit{in vitro} model, serving as a case study that demonstrates rapid adjustments and refinements in data analysis and presentation. Such efficiency is crucial for transparently handling semi-big data \dashedsentence{defined here as manageable, but big enough to cause reproducibility problems} generated by biomedical research.

This thesis illustrates the synergy between experimental methodologies and advanced data analysis tools. The \textit{in vitro} model provides a robust platform for studying myeloma dissemination, while \texttt{plotastic} addresses the need for efficient data analysis. Together, they offer a comprehensive approach to handling complex experiments, advancing both cancer biology and research practices in support of transparent analysis of biological phenomena.

Your draft summary effectively captures the essence of your PhD thesis by emphasizing the integration of biomedical research with data science through practical applications. Here are a few suggestions to make the summary even more concise and impactful:

1. **Clarity and Flow**: Some sentences can be shortened or split to improve readability.
2. **Redundancy**: Consider removing repetitive phrases to enhance the flow and prevent redundancy.
3. **Technical Detail**: Ensure that technical terms and concepts are clearly introduced for readers who might not be familiar with the specifics of the field.

Here's a revised version incorporating these points:

```latex
This PhD thesis integrates biomedical research and data science, focusing on an \textit{in
vitro} model for studying myeloma cell dissemination and a Python-based tool,
\texttt{plotastic}, for semi-automated analysis of multidimensional datasets. The research
addresses two major challenges: understanding the early steps of myeloma dissemination and
improving data analysis efficiency to enhance reproducibility in biomedical research.

In the experimental component, primary human mesenchymal stromal cells (hMSCs) are co-
cultured with INA-6 myeloma cells to study cell proliferation, attachment, and detachment
via time-lapse microscopy. Detachment often follows cell division, predominantly driven by
daughter cells. Novel separation techniques were developed to isolate myeloma
subpopulations for further characterization by RNAseq, cell viability, and apoptosis
assays. The differential expression of adhesion and retention factors, correlated with
patient survival data, offers new insights into myeloma dissemination mechanisms and
potential therapeutic targets.

\texttt{plotastic} enhances the handling of complex data by automating statistical
analyses and integrating with tools like \texttt{seaborn} and \texttt{pingouin} to
streamline visualization and data management. The thesis reflects on the application of
\texttt{plotastic} to the \textit{in vitro} model, illustrating how the software
facilitates rapid adjustments and refines data analysis, addressing semi-big data
challenges – data that are manageable yet complex enough to complicate reproducibility.

Together, the \textit{in vitro} model and \texttt{plotastic} provide a comprehensive
approach to conducting and analyzing complex experiments. This integration advances cancer
biology and biomedical research methodologies, supporting the transparent and precise
analysis of biological phenomena and fostering scientific progress with effective
treatments.
```

This version strives for a tighter narrative by merging related ideas and trimming excess details, which might help in emphasizing the innovative aspects of your work more effectively.

ok, using your input I have drafted this final version!

This PhD thesis integrates biomedical research and data science, focusing on an \textit{in vitro} model for studying myeloma cell dissemination and a Python-based tool, \texttt{plotastic}, for semi-automated analysis of multidimensional datasets. The research addresses two major challenges: (1) understanding the early steps of myeloma dissemination and (2) improving data analysis efficiency to enhance reproducibility in biomedical research.

In the experimental component, primary human mesenchymal stromal cells (hMSCs) are co-cultured with INA-6 myeloma cells to study cell proliferation, attachment, and detachment via time-lapse microscopy. Key findings reveal that detachment often follows cell division, predominantly driven by daughter cells. Novel separation techniques were developed to isolate myeloma subpopulations for further characterization by RNAseq, cell viability, and apoptosis assays. Differential expression of adhesion and retention factors upregulated by INA-6 cells correlates with patient survival. Overall, this work provides insights into myeloma dissemination mechanisms and identifies genes that potentially counteract dissemination through adhesion, which could be relevant for the design of new therapeutics.

To manage complex data, \texttt{plotastic} was developed, a Python-based software that streamlines analysis and visualization of multidimensional datasets. \texttt{plotastic} is built on the idea that statistical analyses are performed based on how the data is visualized. This approach not only simplifies data analysis, but semi-automates analysis in a standardized statistical protocol. The thesis becomes a case study as it reflects on the application of \texttt{plotastic} to the \textit{in vitro} model, demonstrating how the software facilitates rapid adjustments and refinements in data analysis and presentation. Such efficiency is crucial for handling semi-big data transparently, which \dashedsentence{despite being managable} is complex enough to complicate reproducibility.

Together, this thesis illustrates the synergy between experimental methodologies and advanced data analysis tools. The \textit{in vitro} model provides a robust platform for studying myeloma dissemination, while \texttt{plotastic} addresses the need for efficient data analysis. Together, they offer a comprehensive approach to handling complex experiments, advancing both cancer cell biology and research practices in support of transparent analysis of cell biological phenomena.

Your final version of the summary is succinct and effectively communicates the essence of your thesis. It does a great job of connecting the biological research with the development and application of your data analysis tool, \texttt{plotastic}. Here are a few minor suggestions to polish it further:

1. **Clarification and Consistency**: Ensure that technical terms and the software's benefits are clearly explained to maintain consistency throughout the summary.
2. **Flow and Readability**: Adjust sentence structure for smoother transitions between sections.

Here's a slightly revised version with these considerations:

```latex
This PhD thesis integrates biomedical research and data science, focusing on an \textit{in
vitro} model for studying myeloma cell dissemination and a Python-based tool,
\texttt{plotastic}, for semi-automated analysis of multidimensional datasets. The research
tackles two major challenges: understanding the early steps of myeloma dissemination and
enhancing data analysis efficiency to improve reproducibility in biomedical research.

In the experimental component, primary human mesenchymal stromal cells (hMSCs) are co-
cultured with INA-6 myeloma cells to investigate cell proliferation, attachment, and
detachment using time-lapse microscopy. Key observations indicate that cell detachment
frequently follows division, primarily driven by daughter cells. Novel separation
techniques were developed to isolate distinct myeloma subpopulations for further
characterization through RNAseq, cell viability, and apoptosis assays. The differential
expression of adhesion and retention factors correlated with patient survival provides new
insights into myeloma dissemination mechanisms and identifies potential therapeutic
targets.

\texttt{plotastic}, a Python-based software, was developed to manage the complex data from
these experiments. The tool streamlines the analysis and visualization of multidimensional
datasets by ensuring that statistical analyses are closely aligned with how data is
visualized. This method not only simplifies data analysis but also semi-automates it
within a standardized statistical framework. The thesis serves as a case study
demonstrating how \texttt{plotastic} enables rapid adjustments and refinements in data
```

```
    analysis and presentation, crucial for transparently managing semi-big data which,
    although manageable, is complex enough to challenge reproducibility.

    Together, this thesis demonstrates the synergy between experimental methodologies and
    advanced data analysis tools. The \textit{in vitro} model provides a robust platform for
    studying myeloma dissemination, while \texttt{plotastic} addresses the need for efficient
    data analysis. Combined, they offer a comprehensive approach to handling complex
    experiments, advancing both the study of cancer cell biology and research practices in
    support of transparent analysis of biological phenomena.
```

This version refines the language to enhance clarity and ensures that the narrative flows smoothly from describing your research to detailing the application and benefits of \texttt{plotastic}. It maintains a professional tone while clearly outlining the contributions and implications of your work.

Ok! Thank you! I implemented some of your helpful suggestions! Now, please translate this final version into german:

This PhD thesis integrates biomedical research and data science, focusing on an \textit{in vitro} model for studying myeloma cell dissemination and a Python-based tool, \texttt{plotastic}, for semi-automated analysis of multidimensional datasets. The research addresses two major challenges: (1) understanding the early steps of myeloma dissemination and (2) improving data analysis efficiency to enhance reproducibility in biomedical research.

In the experimental component, primary human mesenchymal stromal cells (hMSCs) are co-cultured with INA-6 myeloma cells to study cell proliferation, attachment, and detachment via time-lapse microscopy. Key findings reveal that detachment often follows cell division, predominantly driven by daughter cells. Novel separation techniques were developed to isolate myeloma subpopulations for further characterization by RNAseq, cell viability, and apoptosis assays. Differential expression of adhesion and retention factors upregulated by INA-6 cells correlates with patient survival. Overall, this work provides insights into myeloma dissemination mechanisms and identifies genes that potentially counteract dissemination through adhesion, which could be relevant for the design of new therapeutics.

To manage complex data, a Python-based software named \texttt{plotastic} was developed that streamlines analysis and visualization of multidimensional datasets. \texttt{plotastic} is built on the idea that statistical analyses are performed based on how the data is visualized. This approach not only simplifies data analysis, but semi-automates analysis in a standardized statistical protocol. The thesis becomes a case study as it reflects on the application of \texttt{plotastic} to the \textit{in vitro} model, demonstrating how the software facilitates rapid adjustments and refinements in data analysis and presentation. Such efficiency is crucial for handling semi-big data transparently, which \dashedsentence{despite being managable} is complex enough to complicate reproducibility.

Together, this thesis illustrates the synergy between experimental methodologies and advanced data analysis tools. The \textit{in vitro} model provides a robust platform for studying myeloma dissemination, while \texttt{plotastic} addresses the need for efficient data analysis. Combined, they offer a comprehensive approach to handling complex experiments, advancing both cancer biology and research practices, in support of exploratory and transparent analysis of challenging phenomena.

Here's your PhD thesis summary translated into German:

Diese Doktorarbeit integriert biomedizinische Forschung und Datenwissenschaften und konzentriert sich auf ein \textit{in vitro}-Modell zur Untersuchung der Dissemination von Myelomzellen sowie ein Python-basiertes Werkzeug, \texttt{plotastic}, zur halbautomatisierten Analyse multidimensionaler Datensätze. Die Forschung adressiert zwei Hauptprobleme: (1) das Verständnis der frühen Schritte der Myelomdissemination und (2) die Verbesserung der Effizienz der Datenanalyse zur Steigerung der Reproduzierbarkeit in der biomedizinischen Forschung.

Im experimentellen Teil werden primäre menschliche mesenchymale Stromazellen (hMSCs) mit INA-6-Myelomzellen kokultiviert, um Zellproliferation, Anhaftung und Ablösung mittels Zeitraffer-Mikroskopie zu untersuchen. Zentrale Erkenntnisse zeigen, dass die Ablösung oft auf die Zellteilung folgt und vorwiegend von Tochterzellen angetrieben wird. Neue Trennungstechniken wurden entwickelt, um Myelom-Subpopulationen für weitere Charakterisierungen durch RNAseq, Zellviabilität und Apoptose-Assays zu isolieren. Die differentielle Expression von Adhäsions- und Retentionsfaktoren, die durch INA-6 Zellen hochreguliert werden, korreliert mit dem Überleben der Patienten. Insgesamt liefert diese Arbeit Einblicke in die Mechanismen der Myelomdissemination und identifiziert Gene, die potenziell die Dissemination durch Adhäsion konterkarieren könnten, was für die Entwicklung neuer Therapeutika relevant sein könnte.

Zur Verwaltung komplexer Daten wurde eine Python-basierte Software namens \texttt{plotastic} entwickelt, die die Analyse und Visualisierung multidimensionaler Datensätze optimiert. \texttt{plotastic} basiert auf der Idee, dass statistische Analysen basierend darauf durchgeführt werden, wie die Daten visualisiert werden. Dieser Ansatz vereinfacht nicht nur die Datenanalyse, sondern automatisiert sie auch teilweise in einem standardisierten statistischen Protokoll. Die Arbeit wird zu einer Fallstudie, da sie die Anwendung von \texttt{plotastic} auf das \textit{in vitro}-Modell reflektiert und zeigt, wie die Software schnelle Anpassungen und Verfeinerungen in der Datenanalyse und -präsentation erleichtert. Eine solche Effizienz ist entscheidend für den transparenten Umgang mit Semi-Big-Data, die, obwohl handhabbar, komplex genug sind, um die Reproduzierbarkeit zu erschweren.

Zusammengefasst veranschaulicht diese Dissertation die Synergie zwischen experimentellen Methoden und fortgeschrittenen Werkzeugen der Datenanalyse. Das \textit{in vitro}-Modell bietet eine robuste Plattform für die Untersuchung der Myelomdissemination, während \texttt{plotastic} den Bedarf an effizienter Datenanalyse adressiert. Zusammen bieten sie einen umfassenden Ansatz für die Bearbeitung komplexer Experimente, fördern sowohl die Krebsbiologie als auch die Forschungspraktiken und unterstützen die explorative und transparente Analyse herausfordernder Phänomene.

in the first paragraph I changed one sentence to new version "(2) improving
   data analysis efficiency to address the complexity- and reproducibility
   bottlenecks currently present in biomedical research.", can you translate that sentence into german?

Certainly! Here's the translation of your revised sentence into German:
"(2) die Verbesserung der Effizienz der Datenanalyse, um die derzeit in der biomedizinischen Forschung vorhandenen Engpässe bezüglich Komplexität und Reproduzierbarkeit zu adressieren."

Alright! Now, I want you to remember the complete context of this work, so you can formulate an "Aims" section. It is supposed to be located at the end of the Introduction, so that it reads as if the results have not been presented yet. List individual aims as bullet points formatted in latex

Given the context of your thesis and its focus, an "Aims" section at the end of the Introduction should outline the primary objectives of your research clearly, specifying what you seek to achieve with both the experimental and software development components of your work. Below is a LaTeX formatted "Aims" section that can be incorporated into your thesis:

latex

```latex
\subsection*{Aims of the Study}
This PhD thesis is designed to bridge significant gaps in the understanding and analysis
of myeloma cell behavior and the handling of complex biomedical datasets. The specific
aims are as follows:

\begin{itemize}
    \item Develop an \textit{in vitro} model to elucidate the mechanisms of myeloma cell
dissemination in interaction with mesenchymal stromal cells (hMSCs), focusing particularly
on:
    \begin{itemize}
        \item Observing and quantifying cell proliferation, attachment, and detachment
dynamics using time-lapse microscopy.
        \item Isolating and characterizing distinct myeloma subpopulations interacting
with hMSCs to understand differential gene expression related to cell adhesion and patient
survival.
    \end{itemize}

    \item Design and implement a Python-based software tool, \texttt{plotastic}, to
facilitate the semi-automated analysis of multidimensional datasets generated in
biomedical research. This tool will aim to:
    \begin{itemize}
        \item Streamline the data analysis process, making it more efficient and
reproducible.
        \item Integrate visualization and statistical analysis capabilities to ensure that
data analysis protocols are aligned with the ways in which data is visualized.
        \item Provide a case study demonstrating the application of \texttt{plotastic} in
the analysis of \textit{in vitro} dissemination experiments, emphasizing the tool’s
```

```
ability to handle semi-big data and enhance reproducibility.
    \end{itemize}

    \item Synthesize the findings from the experimental and software development
components to advance the understanding of myeloma dissemination and improve research
practices in biomedical data analysis.
\end{itemize}

These aims are crafted to address both the biological and technical challenges in current
cancer research methodologies and data science applications in biomedicine, fostering
advancements that could lead to novel therapeutic strategies and more robust scientific
inquiries.
```

This section succinctly lays out the research goals, structured to lead into the subsequent chapters that detail the methodologies, results, and significance of your findings.

ChatGPT kann Fehler machen. Überprüfe wichtige Informationen.