

Exposé

Julia zur Verarbeitung und Visualisierung von Daten

Autor: Theodor Schwarzrock
Lehrer: Herr Möller

Seminarkurs Informatik
Friedrich-Schiller-Gymnasium
08.04.2024

Inhaltsverzeichnis

1 Einleitung.....	3
2 Problemstellung.....	3
3 Erkenntnisinteresse.....	3
4 Forschungsstand und Quellenlage	4
5 Forschungsfrage	4
6 Methodik und Forschungsdesign	5
7 Zielsetzung.....	5
8 Vorarbeiten.....	5
9 Gliederungsentwurf	6
10 Zeitplan.....	7
11 Literaturverzeichnis	7

1 Einleitung

Am 14. Februar 2012 wurde eine neue Programmiersprache namens Julia veröffentlicht. Sie wurde entwickelt, um hohe Leistungsfähigkeit mit einer einfachen und effizienten Syntax zu kombinieren. Dabei orientierte man sich an Programmiersprachen wie C, R, Python und Matlab. Ihr Hauptfokus liegt im wissenschaftlichen Rechnen, maschinellen Lernen und der Data-Science.

Data-Science beschreibt unter anderem den Umgang, die Analyse und die Darstellung jeglicher Daten. Sie ermöglicht es, grobe Daten so zu verarbeiten, dass ein Verständnis aus ihnen zu bilden ist. Gerade in unserem Zeitalter der Informationen gewinnt die Data-Science zunehmenden an Bedeutung.

2 Problemstellung

Die Verarbeitung und Visualisierung von Daten mit Hilfe von Programmiersprachen ist durchaus ein relevantes Thema. Denn mit einer stetig zunehmenden Menge an Daten im Alltag und den verschiedensten Branchen, wie Wissenschaft und Forschung, dem Gesundheitswesen, dem Finanzwesen oder der Bildung, ist eine vielseitige Programmiersprache gefragt. Komplexe Angelegenheiten können mit Hilfe einer solchen Programmiersprache intuitiv und verständlich visualisiert werden.

Jene Lücke können zum Beispiel Python oder R füllen. Die beiden Programmiersprachen sind die größten in der Data-Science und ermöglichen durch eine umfangreiche Auswahl an Bibliotheken und Paketen die Verarbeitung und Visualisierung von Daten. Als moderne Sprache versucht Julia deren besten Eigenschaften zu vereinen.

Es stellt sich die Frage, ob dies Julia gelingt und sie neue Möglichkeiten in der Data-Science eröffnet, welche die anderen Sprachen nicht bieten können. Dafür werden vor allem das mathematische Rechnen und die Veranschaulichung der Ergebnisse in Betracht gezogen.

3 Erkenntnisinteresse

Schon immer habe ich mich für Naturwissenschaften interessiert. Vor allem in der Mathematik und Physik konnte ich Begeisterung finden. Als ich mich dann während der Corona-Pandemie mit der Informatik befasst habe, war ich sofort gefesselt. Schnell habe ich mir JavaScript und Python angeeignet, wobei Python mein klarer Favorit war.

Eine Schnittstelle dieser Wissenschaften habe ich in der Data-Science gefunden. Durch geeignete Darstellung von physikalischen Phänomenen oder mathematischen Sachverhalten kann man diese weitaus besser verstehen. So habe ich zum Beispiel ein Programm, welches das n-Körper-Problem visualisiert, geschrieben, sodass ich die Parameter nach meinem Belieben verändern konnte.

Dadurch konnte ich mein Verständnis und meine Intuition zu dem Thema deutlich vergrößern.

Nun möchte ich untersuchen, ob sich Julia für solche Angelegenheiten vergleichbar oder vielleicht sogar mehr eignet.

4 Forschungsstand und Quellenlage

Da Julia sich stark an Python orientiert gibt es reichlich Artikel, welche die beiden Programmiersprachen auf jeglichen Ebenen vergleichen. Zu einen solchen Artikel habe ich im Sinne des Seminarkurses habe ich ein Expert geschrieben. Der Artikel „The brass tacks of Julia programming“ [1] konzentriert sich besonders auf grundlegende Befehle und bietet direkt einen Vergleich zu Python. Um einen detaillierteren Einblick auf Julias Syntax zu erhalten, könnte sich das Einsteigerbuch „Think Julia“ [2] eignen.

Einen Überblick auf Julia in der Data-Science können die Bücher „Julia for Data Analysis“ [3] und „Statistics with Julia“ [4] darstellen. Diese gehen, wie ihre Namen es schon vorwegnehmen, sehr umfangreich auf Julias Umgang mit Daten ein. Da die Datenvisualisierung im Fokus der Seminararbeit liegt, wird sich das Buch „Interactive Visualization and Plotting with Julia“ [5] sicherlich nützlich machen können. Dort findet man spezifisch die Implementation der Visualisierung mit Hilfe verschiedener Bibliotheken.

Diese Sammlung an Quellen sollten eine solide Basis für die praktischen Grundlagen zur Erarbeitung der Seminararbeit bieten. Für den Vergleich mit Python gibt es, wie bereits erwähnt, ausreichend Artikel. Auch hin bezüglich der Data-Science.

5 Forschungsfrage

Die eher grobe Leitfrage, wie sich Julia im wissenschaftlichen Rechnen und der Data-Science verhält, kann man in mehrere Forschungsfragen unterteilen. Zunächst ist es sinnvoll, sich zu fragen, warum Julia in diesen Bereichen heraussticht. Welche Besonderheiten besitzt Julia, die die Programmiersprache besonders machen?

Um Julias Platz unter den vielen Programmiersprachen zu etablieren, wird ein Vergleich mit Python vorgenommen. Dieser sollte auf verschiedenen Ebenen stattfinden, wobei gleichzeitig die Eigenschaften von Julia erarbeitet werden. Leitfragen hierbei können sein: Wie unterscheiden sich Julias und Pythons Syntax? Macht sich bemerkbar, dass Julia auf Python basiert? Welche der beiden Sprachen ist schneller (hin bezüglich der relevanten Teilgebiete)? Welche Sprache hat die größere Standardbibliothek? Können Julias Bibliotheken die vielen von Python konkurrieren?

Mit Hilfe unter anderem dieser Erkenntnisse lässt sich schließlich die Notwendigkeit Julias bewerten.

6 Methodik und Forschungsdesign

Der praktische Teil der Seminararbeit, also das Programmieren mit Julia, widmet sich der Visualisierung des Überganges zwischen den fünf platonischen Körpern. Hierzu werden mathematische Sachverhalte und das benötigte Wissen über Julia vermutlich durch Sekundärquellen aus dem Internet erarbeitet, da diese nicht das zentrale Thema der Seminararbeit sind. Falls Kenntnisse über Python für den Vergleich fehlen, werden diese ebenso herangezogen. Das Programmieren wird in Visual-Studio-Code stattfinden, da dies eingebaute Erweiterung besitzt, die die Programme intern anzeigen lässt.

In den theoretischen Teil sollen zum einem die Erfahrungen aus dem praktischen Teil benutzt und referenziert werden und zum anderen neue Kenntnisse aus Primärquellen eingebracht werden. Dabei geht es darum, möglichst objektiv und faktenorientiert zu arbeiten.

7 Zielsetzung

Das Ziel dieser Seminararbeit ist es, sich mit der Programmiersprache Julia bekannt zu machen und ihre Fähigkeiten im mathematischen Rechnen und dem Visualisieren der entstehenden Daten zu untersuchen.

Wie schon erwähnt, solle gezeigt werden, wie die platonischen Körper zusammenhängen. Dies wird in einer Animation geschehen, wobei der Nutzer auswählen kann, welche Körper angezeigt werden sollen. Ziel ist es zudem, ein möglichst identisches Programm in Python schreiben, sodass ein Seite-an-Seite-Vergleich möglich ist.

Außerdem soll ein Vergleich mit Python stattfinden, da dies die meistgenutzte Programmiersprache in diesen Bereichen ist. Aufgrund dieses Vergleichs und weiteren ausgearbeiteten Fakten über Julia wird schließlich bewertet, ob Julia eine Alternative bzw. einen Ersatz gegenüber anderen Programmiersprachen in den untersuchten Teilbereichen darstellt.

8 Vorarbeiten

Da die hinterlegende Mathematik nicht allzu komplex ist, sind nötige mathematischen Kenntnisse für den praktischen Teil schon vorhanden. Auch beim Arbeiten mit Julia sollte es zu wenigen Problemen kommen, da ich für den Seminarkurs schon einige Programme in Julia geschrieben habe.

Neben dem, im Seminarkurs verfasstem, Exzerpt habe ich bisher kaum Quellenarbeit geleistet. Relevante Quellen sind zwar bekannt, wurden aber noch nicht detailliert analysiert und eingeordnet. Der Großteil dieser Literatur ist jedoch im Internet zugänglich, sodass man sich jederzeit mit ihr auseinandersetzen kann.

9 Gliederungsentwurf

1 Einleitung

2 Programmiersprache Julia

2.1 Entstehung und Geschichte

2.2 Anwendungen

2.3 Konkurrenten

3 Praktische Anwendung von Julia

3.1 Verwendung von Julia und Python

3.2 Unterschiede und Gemeinsamkeiten

3.3 Subjektive Auswertung

4 Vergleich mit Python

4.1 Syntax und Semantik

4.2 Leistung

4.3 Community und Unterstützung

5 Fazit

6 Literaturverzeichnis

10 Zeitplan

Inhalt	Zeitabschnitt
<ul style="list-style-type: none">- Quellen erarbeiten, analysieren und sortieren- nötige Literatur beschaffen	April bis Mai
<ul style="list-style-type: none">- Abschnitt 1 und 2 auf Papier bringen	Mai bis Juni
<ul style="list-style-type: none">- Programme in beiden Programmiersprachen schreiben- Abschnitt 3 auf Papier bringen	Juni bis August
<ul style="list-style-type: none">- Abschnitt 4 und 5 auf Papier bringen	August bis Oktober
<ul style="list-style-type: none">- Formatieren, Korrekturlesen- letzte Überarbeitungen	Oktober

11 Literaturverzeichnis

- [1] Tayal, Anshu (03.12.2021) „The brass tacks of Julia programming“
URL: <https://blog.quantinsti.com/julia-syntax/>
[abgerufen am 06.04.2024]
- [2] Lauwrens, Ben / Downey, Allen B. (05.04.2019) „Think Julia“
URL: <https://www.oreilly.com/library/view/think-julia/9781492045021/>
[abgerufen am 06.04.2024]
- [3] Kamiński, Bogumił (03.01.2023) „Julia for Data Analysis“
URL: <https://www.manning.com/books/julia-for-data-analysis>
[abgerufen am 06.04.2024]
- [4] Nazarathy, Yoni / Klok, Hayden (04.09.2021) „Statistics with Julia“
URL: <https://link.springer.com/book/10.1007/978-3-030-70901-3>
[abgerufen am 06.04.2024]
- [5] Zea, Diego Javier (29.08.2022) „Interactive Visualization and Plotting with Julia“
URL: <https://www.packtpub.com/product/interactive-visualization-and-plotting-with-julia/9781801810517>
[abgerufen am 06.04.2024]