Was ist Optimierung?

Optimierung ist die Minimierung oder Maximierung einer linearen oder nicht linearen Zielfunktion mit einer endlichen Anzahl an unbekannten Parametern und einer oder mehreren Constrains (siehe Kapitel Was sind Constrains?).

Eine lineare Zielfunktion ist eine Polynomfunktion höchstens ersten Grades und folgt einem festen Schema (siehe Minitab, 2017). Jeder Parameter wird mit Koeffizienten multipliziert und aufsummiert und optional mit einer Konstante addiert (siehe Formel 1).

1

Dabei haben die Konstanten und Koeffizienten einen festen, frei wählbaren Wert während die Parameter innerhalb der Funktion nicht verändert werden dürfen. Eine nicht Lineare Zielfunktion entspricht dann allem, was nicht in dieses Schema passt. Die meisten realen Probleme die sich durch Optimierung lösen lassen können mit linearen Zielfunktionen gelöst werden, weshalb nur auf die lineare Optimierung näher eingegangen wird. Bei der Optimierung wird bei einer bekannten Anzahl an Parametern eine Zielfunktion mit geeigneten Constrains definiert, für die dann Parameter gesucht werden, die die Zielfunktion entweder minimieren oder maximieren (siehe ).

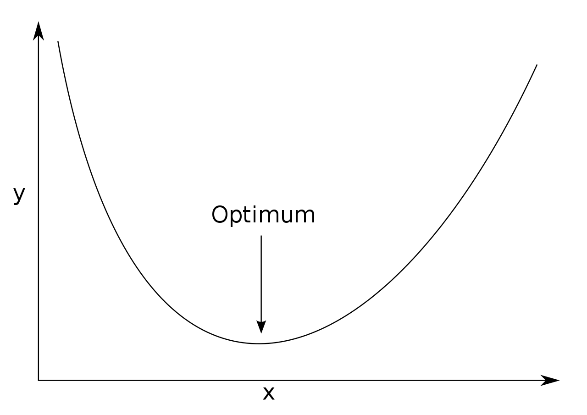
(Nickel, et al., 2022, pp. 8-10)

Abbildung 1: Minimum einer nicht linearen Zielfunktion dargestellt in einem Graphen

<https://de.wikipedia.org/wiki/Optimierung_(Mathematik)#/media/Datei:Optimum.svg>

Ein Beispiel für eine lineare Zielfunktion könnte wie folgt aussehen:

Beispiel 1: Explizites Beispiel einer linearen Zielfunktion

So wie diese Funktion definiert ist kann aber noch kein Optimum gefunden werden. Wenn ein Maximum gesucht wird kann man unendlich große Zahlen als Parameter angeben und bei einem Minimum unendlich kleine. Deshalb muss mindestens für jeden Parameter eine geeignete Constrains definiert werden.

Was sind Constrains?

Constrains können Einschränkungen der Parameter im Zahlenbereich, oder zusätzliche Ungleichungen sein, die erfüllt werden müssen um die Zielfunktion gültig zu optimieren. Die Anzahl der Ungleichungen ist beliebig und beinhalten eine oder mehrere festgesetzte Parameter der Zielfunktion. Bei der linearen Optimierung müssen diese Ungleichungen ebenfalls linear sein (siehe Was ist Optimierung?).

Um das Problem im zu lösen muss für jeden Parameter eine Ungleichung erstellt werden, die den jeweiligen Parameter einschränkt.

Beispiel 2: Mögliche einschränkende Constrains von Parametern einer Zielfunktion

Jetzt kann sowohl ein Maximum als auch ein Minimum berechnet werden.  
Um Reelle Zahlen zu vermeiden muss nur der Zahlenbereich der Parameter auf Natürliche Zahlen angepasst werden (siehe ).

Beispiel 3: Mögliche Einschränkung des Zahlenbereiches von Parametern einer Zielfunktion

Ungleichungen können auch komplexer definiert werden:

Beispiel 4: Beispiel einer komplexen Nebenbedingung

Ungeeignete Constrains

Die Constrains sollten verhindern, dass für Parameter unendliche Werte eingesetzt werden können, da diese in realen Anwendungen nicht relevant sind.   
Außerdem sollten die Ungleichungen so definiert werden, dass die Zielfunktion immer gelöst werden kann. In wird definiert, dass  0 0 oder 1 sein muss und zu gleich soll  0 \* 4 zwischen 2 und 3 liegen. Das führt zu einer nicht lösbaren Zielfunktion. Dies mag hier offensichtlich erscheinen, stellt aber durchaus ein Problem bei der dynamischen Generierung einer Zielfunktion mit Constrains in einem Programm dar.

Beispiel 5: Beispiel unlösbarer Constrains

Lösung des Zuweisungsproblems

Das beschriebene Zuweisungsproblem (siehe Kapitel Leon) ist ein bekanntes und weit verbreitetes Problem in der Optimierung und fällt unter die Kategorie Operations Research (OR). „Operations Research (OR) bezeichnet einen Wissenszweig, der sich mit der Analyse von praxisnahen, komplexen Problemstellungen im Rahmen eines Planungsprozesses zum Zweck der Vorbereitung von möglichst guten Entscheidungen durch die Anwendung mathematischer Methoden beschäftigt. Die Hauptaufgaben im OR bestehen in der Abbildung eines realen Entscheidungsproblems durch ein Optimierungs- oder Simulationsmodell […] und die Anwendung bzw. Entwicklung eines Algorithmus zur Lösung des Problems. Dabei spielt Software-Unterstützung eine zentrale Rolle […].“ (Domschke, et al., 2015, p. 1) Dabei lässt sich das Problem in eine lineare Zielfunktion und linearen Ungleichungen als Constrains beschreiben.

Optimierung in Computeranwendungen

Das bisher gezeigte ist nur ein grober Überblick wie Optimierung funktioniert. Eine eigene Implementierung einer Anwendung ohne sich in diesem Bereich spezialisiert zu haben ist dementsprechend wenig sinnvoll. Damit sich Programmierer trotzdem solchen Anwendungen widmen können werden von hoch qualifizierten und spezialisierten Fachkräften Frameworks zu Verfügung gestellt. „In der Softwareentwicklung ist ein Framework ein Entwicklungsrahmen, der dem Anwendungsprogrammierer zur Verfügung steht, um die grundlegende Architektur der Software zu bestimmen.“ (Business Systemhaus, o. D.) Für die Optimierung gibt es ebenfalls Frameworks für verschiedene Anwendungsbereiche. Dabei stellen diese ein vereinfachtes Interface zur Verfügung, das die Komplexität der Optimierung versteckt. Das hat zur Folge, dass sich Programmierer, die nicht auf Optimierung spezialisiert sind, der weniger komplexen Datenvorbereitung für das Framework widmen können.

Welches Framework wurde verwendet?

Es wurde das Framework OR-Tools von Google verwendet.

Für die Auswertung eines geeigneten Frameworks gab es folgende Voraussetzungen in absteigender Wichtigkeit:

* geeignet für OR, speziell dem Zuweisungsproblem
* Hohe Performance bei geringen Verhältnis FTS zu Aufgabe
* Programmiersprache: C#

Es wurde sich an der MiniZinc Challenge orientiert. „Die MiniZinc Challenge ist ein jährlicher Wettbewerb von Constraint-Programming-Lösern zu einer Vielzahl von Benchmarks. Er findet seit 2008 jedes Jahr statt, wobei die Gewinner auf der jährlichen International Conference on Principles and Practice of Constraint Programming bekannt gegeben werden.” (übersetzt aus Monash University, o. D.). In dieser Challange erziehlten die OR-Tools die besten Ergebnisse für die letzten Jahre (siehe Monash University, o. D.). Mit dieser Leistung ist dieses Framework mit Abstand das Beste auf dem Markt (Stand 30.12.2022 nach MiniZinc, Monash University). Außerdem ist sie in verschiedenen Programmiersprachen erhältlich wie Python, Java und C# (siehe Google Startleitfäden, o. D.) und ist eine Open-Source-Software (siehe Google OR-Tools, o. D.). Diese Eigenschaften heben die OR-Tools deutlich von den anderen Frameworks ab.

# Literaturverzeichnis

Business Systemhaus, o. D.. *bsh-ag.* [Online]   
Available at: https://bsh-ag.de/it-wissensdatenbank/framework/  
[Zugriff am 28 Dezember 2022].

Domschke, W., Drexl, A., Klein, R. & Scholl, A., 2015. *Einführung in Operations Research.* 9. Auflage Hrsg. Berlin, Deutschland: Springer Berlin Heidelberg.

Google, o. D.. *developers.google.* [Online]   
Available at: https://developers.google.com/optimization/introduction/get\_started  
[Zugriff am 30 Dezember 2022].

Google, o. D.. *developers.google.* [Online]   
Available at: https://developers.google.com/optimization/introduction/overview  
[Zugriff am 30 Dezember 2022].

Minitab, 2017. *blog.minitab.* [Online]   
Available at: https://blog.minitab.com/en/adventures-in-statistics-2/what-is-the-difference-between-linear-and-nonlinear-equations-in-regression-analysis  
[Zugriff am 28 Dezember 2022].

Monash University, o. D.. *minizinc.* [Online]   
Available at: https://www.minizinc.org/challenge.html  
[Zugriff am 30 Dezember 2022].

Nickel, S., Rebennack, S., Stein, O. & Waldmann, K.-H., 2022. *Operations Research.* 3. Auflage Hrsg. Berlin, Deutschland: Springer Berlin Heidelberg.