

# Dienstplanoptimierung

## Version 4

Stefan Mann und Ruslan Krenzler

22. März 2020

## 1 Einführung

Einen Dienstplan für einen Krankenhaus manuell zu erstellen ist schwierig. In einer Krisensituation, wenn Personal und Zeit knapp sind und die Situation sich ständig ändert, ist das noch schwieriger. Um das Krankenhaus bei dieser Aufgabe zu unterstützen, haben wir verschiedene Algorithmen entworfen. Im Besonderen haben wir die aktuelle Lage der Covid19 Pandemie berücksichtigt.

Ziel dieser Arbeit ist es ein interaktives Verfahren zur Erstellung eines Dienstplanes zur Verfügung zu stellen. Interaktiv heißt, der Personalverantwortliche bekommt nicht einfach nur einen automatisch generierten Dienstplan als Lösung präsentiert sondern kann die Auswahl vollständig kontrollieren. Dabei wird er durch ein intelligentes Scoring System unterstützt, welches ihm hilft gute Entscheidungen zu treffen.

Wir wollen uns dabei eng an ein konkretes Beispiel aus dem Krankenhausalltag halten und dort die Monatsplanung der Pfleger einer Station modellieren. Dies ist eine deutlich abgespeckte Version des allgemeinen Problems, jedoch zur Skizzierung der Gedanken geeignet. Die folgenden Ideen und Anregungen sind innerhalb von 48h während des #WirVsVirus-Hackathon2020 entstanden und verfasst worden. Sie müssen noch genau überprüft und getestet werden.

### Die Herausforderungen

- optimale Dienstpläne zu berechnen ist sehr schwer (NP-schwer)
- es gibt keine Garantie, dass eine optimale Lösung in der Realität funktionieren wird, da kurzfristige Ausfälle die gesamte Planung durcheinander bringen kann
- aufgrund einer Notsituationen erwarten wir sogar Verletzungen von Regeln, wie Arbeitszeit, Ruhephasen und Personalmangel
- aufgrund zu vieler Parameter fällt es schwer den Überblick zu behalten und gute Lösungen zu finden. Wir müssen gleichzeitig unterschiedliche Prioritäten berücksichtigen: Überstunden, Personalmangel, Ansteckungsgefahr,

Belastung der Mitarbeiter. - am Ende möchte der Mensch das letzte Wort haben und den Plan nach seinen Wünschen gestalten

- Regelungen und Wünsche zu formalisieren ist ein langwieriger Prozess

## Szenario

Eine Personalplanerin versucht für den kommenden Monat geeignete Mitarbeiter in ihr bestehendes Schichtmodell einzutragen. Sie möchte Arbeitszeitenregelungen, Ruhephasen, Wünsche der Mitarbeiter, und potentiellen Covid-Status des Mitarbeiters berücksichtigen

## 2 Iterativer Algorithmus

### Idee

Wir nutzen ein Score System. Jeder noch nicht eingeplanter Mitarbeiter, erhält einen Score zwischen 0 und 1. Dieser Score berücksichtigt Arbeitszeiten, Ruhephasen, Ansteckungsgefahr und zukünftigen Personalstand. Je höher der Score ist, desto besser ist der Mitarbeiter für die jeweilige Schicht geeignet.

Dieser Score wird aus Gründen der Anschaulichkeit in ein Ampelsystem umgewandelt, so dass die Personalerin schnell die geeignetsten Kandidaten sowie Grenzfälle sehen kann. Sie wählt einen Kandidaten und das System berechnet Scores für restlichen Mitarbeiter erneut. Schritt für Schritt füllt sie interaktiv gestützt die Pläne aus.

### Algorithmische Realisierung

Wir nennen den Score, der im Ampelsystem dargestellt wird, einen Gesamtscore  $S_{\text{total}}$ . Er setzt sich zusammen aus thematisch getrennten Teilscores. Diese Teilscores sind auch Werte zwischen 0 und 1. Zusammenaddiert und gewichtet ergeben sie den Gesamtscore.

Wir nutzen folgende Teilscores: Arbeitszeiten-Score  $S_{\text{time}}$ , COVID-19-Score  $S_{\text{COVID}}$ , Fluktuations-Score  $S_{\text{team}}$  und Mitarbeiterwünsche-Score  $S_{\text{wish}}$ .

### Arbeitszeiten-Score

Der Arbeitszeiten-Score  $S_{\text{time}}$  beschreibt, ob bzw. wie gut die relevanten Arbeitszeiten- und Freischichtenregelungen eingehalten werden. Er ist Null, falls aus Arbeitsrechtlichen Gründen eine Einteilung des Mitarbeiters nicht möglich ist. Er ist 1, falls eine Einteilung problemlos stattfinden kann. Je höher der Score desto eher steht der Mitarbeiter zur Verfügung. Mit folgender Darstellung bilden wir das gewünschte Verhalten ab:

$$S_{\text{time}} := \begin{cases} 1 - \frac{\text{current\_hours}}{\text{max\_hours}} & \text{gesetzliche Vorgaben erfüllt,} \\ 0 & \text{gesetzliche Vorgaben verletzt.} \end{cases}$$

In dieser Formel ist *current\_hours* die Anzahl von bereits geleisteten Arbeitsstunden und *max\_hours* ist die Anzahl der Arbeitsstunden, die der Mitarbeiter insgesamt machen darf.

Überstunden können durch Erhöhung des Wertes *max\_hours* berücksichtigt werden.

### COVID-19-Score

Der Covid19\_Score  $S_{\text{COVID}}$  spiegelt die Wahrscheinlichkeit wieder, dass der Mitarbeiter sich bei Schichtübernahme nicht mit Covid19 infiziert. Er ist 1, wenn wir 100%-tiger Sicherheit sagen können, der Mitarbeiter ist infiziert. Er ist 0, wenn wir mit 100%-tiger Sicherheit sagen können, der Mitarbeiter ist nicht infiziert. Wir nutzen dazu die Formel

$$S_{\text{COVID}} := (1 - p \cdot r)^n \cdot (1 - r)^k.$$

- $p$  ist die Wahrscheinlichkeit dass ein Kontakt Covid hat.
- $r$  ist die Wahrscheinlichkeit von einer infizierten Person angesteckt zu werden.
- $n$  ist die Anzahl von Kontakten mit Personen mit unbekanntem Covid-Status.
- $k$  ist die Anzahl von Kontakten mit Personen mit Covid.

Diese Formel basiert auf einem Wahrscheinlichkeitsmodell im Kapitel 4.

### Fluktuationscore

Der Fluktuationscore  $S_{\text{team}}$  soll die Präferenz für konstante Teams während der Planung berücksichtigen. Er ist sehr klein, falls der Mitarbeiter mit vielen Kollegen in Kontakt stand während der Planung. Er ist jedoch nie niemals Null, da er keinen entscheidenden Einfluss auf den Gesamtscore haben darf. Er ist 1, falls der Mitarbeiter mit noch keinem weiteren Kollegen zusammengearbeitet hat.

$$S_{\text{team}} := \frac{\text{employees} - \text{contacts}}{\text{employees}}$$

*employees* ist die Gesamtanzahl aller Mitarbeiter, *contacts* ist die Gesamtzahl aller Kontakte zu anderen Mitarbeitern im derzeitigen Plan

Bemerkung: Da ein Mitarbeiter nicht mit sich selbst in Kontakt steht, folgt Fluktuationscore ungleich Null.

### Mitarbeiterwünsche-Score

Der Mitarbeiterwunsch-Score  $S_{\text{wish}}$  berücksichtigt den Wunsch des Mitarbeiter an einer Schicht zu arbeiten oder nicht zu arbeiten. Es ist eine Zahl zwischen 0 und 1. Der Wert 1 bedeutet, dass der Mitarbeiter an dieser Schicht arbeiten möchte, der Wert 0 bedeutet, dass der Mitarbeiter an dieser Schicht lieber nicht an dieser Schicht arbeiten möchte. Dieser Wert muss manuell eingetragen werden.

### Gesamtscore

Der Gesamtscore  $S_{\text{total}}$  fasst Teilscores zusammen und berücksichtigt deren Wichtigkeit.

$$S_{\text{total}} := \frac{W_{\text{time}}S_{\text{time}} + W_{\text{COVID}}S_{\text{COVID}} + W_{\text{team}}S_{\text{team}} + W_{\text{wish}}S_{\text{wish}}}{W_{\text{time}} + W_{\text{COVID}} + W_{\text{team}} + W_{\text{wish}}}$$

Die Gewichtungen  $W_{\text{time}}$ ,  $W_{\text{COVID}}$ ,  $W_{\text{team}}$  und  $W_{\text{wish}}$  werden vom Anwender bestimmt.

## 3 Andere Algorithmen

Man kann auch versuchen, diesen Dienstplan mit Hilfe von evolutionären Algorithmen zu optimieren. Dazu wird jeder persönliche Plan als eine Folge von Zahlen 0, 1, 2 oder 3 dargestellt: 0 bedeutet frei, 1 bedeutet Frühschicht, 2 bedeutet Spätschicht und 3 bedeutet Nachtschicht. Ein Dienstplan besteht aus einer Folge von persönlichen Plänen. Ein Dienstplan für drei Tage und vier Personen kann zum Beispiel so aussehen:

[3,3,0, 1,1,0, 2,2,1, 1,1,0]

Dieser Plan bedeutet: Der erste Mitarbeiter arbeitet zwei Nachgeschichten und hat dann frei, der zweite Mitarbeiter arbeitet zwei Frühschichten und dann hat frei und so weiter.

Der evolutionäre Algorithmus benötigt eine Kostenfunktion. Wir haben dazu die Anzahl von fehlenden Mitarbeitern in dem gesamten Plan gewählt und Extrakosten für Verletzung von Arbeitsregeln addiert.

Der evolutionäre Algorithmus wird dann versuchen den Dienstplan so zu ändern, dass die Gesamtkosten niedriger sind. In unserem Fall bedeutet es, dass die Anzahl der fehlenden Mitarbeiter klein ist und möglichst wenige Regeln verletzt werden.

Um die Entscheidungen von Personalplanern mitzuberücksichtigen, kann man dem Algorithmus erlauben, nur bestimmte Teile vom Dienstplan zu ändern.

Wir haben diesen Code mit 30 Tagen, 40 Mitarbeitern und 6 Schichten für ein Szenario mit einfachen Anwendungsregeln getestet

Der Code steht unter LGPL oder GPL zur Verfügung.

## 4 Wahrscheinlichkeitsmodell für Covid-Score

Eine Pflegerin hat direkte Kontakte zu  $n$  Personen und indirekte Kontakte zu  $m$  Personen mit unbekannten Covid-19 Statusen. Sie hat dazu noch Kontakte zu  $k$  Personen mit nachgewiesenen Covid-19. Wie hoch ist die Wahrscheinlichkeit, dass die Pflegerin gesund bleibt.

**Mathematisches Modell**  $G \in \{0, 1\}$  ist eine Zufallsvariable. Sie beschreibt Gesundheitszustand von Pflegerin.  $G = 1$  bedeutet die Pflegerin ist gesund.  $G = 0$  bedeutet die Pflegerin ist infiziert.

$X_i \in \{0, 1\}$  ist eine Zufallsvariable. Sie beschreibt, ob der  $i$ -te direkte Kontakt infiziert ist.  $X_i = 1$  bedeutet, dass direkter Kontakt  $i$  gesund ist.  $X_i = 0$  bedeutet, dieser Kontakt ist infiziert.

$Y_i \in \{0, 1\}$  ist eine Zufallsvariable. Sie beschreibt, ob  $A$  von dem  $i$ -ten Kontakt mit Unbekannten Status infiziert wurde.  $Y_i = 1$  bedeutet keine Infektion.

$Z_i \in \{0, 1\}$  ist eine Zufallsvariable. Sie beschreibt, ob  $Y$  von einem  $i$ -ten Kontakt mit Covid infiziert wurde.  $Z_i = 1$  bedeutet keine Infektion.

Wir bezeichnen die Wahrscheinlichkeit Covid zu haben  $P(X_i = 0) =: p$ .

Wir bezeichnen die Wahrscheinlichkeit von einer infizierte Person angesteckt zu werden  $P(Y_i = 0 | X_i = 0) =: r$ . Weil  $Z_i$  Covid hat, gilt auch  $P(Z_i = 0) = r$ . Ansteckung von einer gesunden Person ist nicht möglich.

Wir nehmen an, dass Infizierungen unabhängig von einander und gleich wahrscheinlich sind (i.i.d). Dann ist die Wahrscheinlichkeit von einer Person  $i$  mit einem unbekannten Covid-Status eingesteckt zu werden

$$\begin{aligned} P(Y_i = 0) &= P(Y_i = 0 \cap X_i = 0) + \underbrace{P(Y_i = 0 \cap X_i = 1)}_{=0(\text{unmöglich})} \\ &= P(Y_i = 0 | X_i = 0)P(X_i = 0) = p \cdot r \end{aligned}$$

Die Wahrscheinlichkeit, dass keine Ansteckung nach diesem Kontakt statt findet, ist

$$P(Y_i = 1) = 1 - p \cdot r.$$

Wegen Unabhängigkeitsannahmen ist die Wahrscheinlichkeit gesund zu bleiben nach  $n$  Kontakten mit Personen mit unbekannten Covid-Status:

$$P(Y_1 = 1 \cap \dots \cap Y_n = 1) = P(Y_1 = 1) \cdot P(Y_n = 1) = (1 - p \cdot r)^n.$$

Nach  $k$  Kontakten mit Personen mit Covid ist die Wahrscheinlichkeit gesund zu bleiben

$$P(Z_1 = 1 \cap \dots \cap Z_k = 0) = (1 - r)^k.$$

Wenn man sowohl unbekannte Kontakte als auch Covid-Kontakte berücksichtigt, bleibt die Pflegerin nur dann gesund, wenn alle Ansteckungsversuche fehlgeschlagen sind. Dann ist

$$\{G = 1\} = \{Y_1 = 1 \cap \dots \cap Y_n = 1 \cap Z_1 = 1 \cap \dots \cap Z_k = 0\}.$$

Somit ist wegen Unabhängigkeit

$$P(G = 1) = P(Y_1 = 1 \cap \dots \cap Y_n = 1) \cdot P(Z_1 = 1 \cap \dots \cap Z_k = 0) = (1-p \cdot r)^n \cdot (1-r)^k.$$

Indirekte Kontakte müssen in diesem Modell nicht berücksichtigt werden.

## 5 Ausblick

Die Modellierung des Dienstplanproblems mit Scoring System ist eine flexible Möglichkeit sich diesem Problem-Typus zu nähern. Im Rahmen der kurzen Arbeitszeit sind viele Fragen offen geblieben. Beispielsweise bleibt zu Überprüfen, inwiefern die gefundenen Lösungen den Anforderungen eines Anwenders entsprechen. Im Speziellen spielt hierbei die Wichtung der einzelnen Teilscores eine entscheidende Rolle und bleibt der Beforschung überlassen. Beim Entwickeln neuer Teilscores für problemspezifische Anpassungen, sind der Kreativität keine Grenzen gesetzt.