EMS Staff Scheduling Optimization Project

Jaipreet Kaur

December 01, 2025

1. Introduction

This project develops a mathematical optimization model to create a minimum-cost weekly schedule for EMS staff.

2. Data

```
staff <- read.csv("staff.csv")
req <- read.csv("requirements.csv")
staff
```

```
##    staff_id     name      type weekday_wage weekend_wage
## 1        P1   Garcia Paramedic          250          300
## 2        P2    Smith Paramedic          250          300
## 3        P3  Johnson Paramedic          250          300
## 4        P4 Williams Paramedic          250          300
## 5        E1    Brown       EMT          180          220
## 6        E2    Jones       EMT          180          220
## 7        E3   Miller       EMT          180          220
## 8        E4    Davis       EMT          180          220
## 9        E5   Wilson       EMT          180          220
## 10       E6    Moore       EMT          180          220
## 11       E7   Taylor       EMT          180          220
## 12       E8 Anderson       EMT          180          220
```

```
req
```

```
##         day day_num min_staff min_paramedics
## 1    Monday       1         6              2
## 2   Tuesday       2         6              2
## 3 Wednesday       3         8              3
## 4  Thursday       4         8              3
## 5    Friday       5        10              4
## 6  Saturday       6         8              3
## 7    Sunday       7         6              2
```

3. Manual Feasible Schedule

```
manual_schedule <- matrix(
  c(
    1,1,1,1,0,0,0,
    1,1,1,1,1,0,0,
    1,0,1,0,1,1,0,
    0,0,1,1,1,1,1,
    1,1,1,1,1,0,0,
    1,1,0,1,0,0,0,
    1,1,1,1,1,1,0,
    1,1,1,1,1,1,0,
    1,0,1,0,1,1,0,
    1,1,1,1,1,1,1,
    1,1,1,1,1,0,0,
    1,1,1,1,1,1,1
  ),
  nrow = 12, byrow = TRUE
)

rownames(manual_schedule) <- staff$staff_id
colnames(manual_schedule) <- req$day
manual_schedule
```

```
##    Monday Tuesday Wednesday Thursday Friday Saturday Sunday
## P1      1       1         1        1      0        0      0
## P2      1       1         1        1      1        0      0
## P3      1       0         1        0      1        1      0
## P4      0       0         1        1      1        1      1
## E1      1       1         1        1      1        0      0
## E2      1       1         0        1      0        0      0
## E3      1       1         1        1      1        1      0
## E4      1       1         1        1      1        1      0
## E5      1       0         1        0      1        1      0
## E6      1       1         1        1      1        1      1
## E7      1       1         1        1      1        0      0
## E8      1       1         1        1      1        1      1
```

```
cost_manual <- function(schedule) {
  total <- 0
  for (i in 1:nrow(schedule)) {
    for (d in 1:ncol(schedule)) {
      wage <- if (d %in% c(6,7)) staff$weekend_wage[i] else staff$weekday_wage[i]
      total <- total + wage * schedule[i, d]
    }
  }
  total
}

manual_cost <- cost_manual(manual_schedule)
manual_cost
```

```
## [1] 12670
```

4. Optimization Model Implementation

```
library(lpSolve)
```

```
## Warning: package 'lpSolve' was built under R version 4.3.3
```

```
staff_ids <- staff$staff_id
days <- req$day_num
n_staff <- length(staff_ids)
n_days <- length(days)

col_index <- function(i,d) (i-1)*n_days + d
n_vars <- n_staff * n_days

# Objective vector
c_vec <- rep(0,n_vars)
for(i in 1:n_staff){
  for(d in 1:n_days){
    wage <- if(d %in% c(6,7)) staff$weekend_wage[i] else staff$weekday_wage[i]
    c_vec[col_index(i,d)] <- wage
  }
}

A <- list(); dir <- c(); rhs <- c()

# Daily staff req
for(d in 1:n_days){
  row <- rep(0,n_vars)
  for(i in 1:n_staff) row[col_index(i,d)] <- 1
  A <- append(A,list(row)); dir <- c(dir,">="); rhs <- c(rhs, req$min_staff[d])
}

# Paramedics
paramedics <- which(staff$type=="Paramedic")
for(d in 1:n_days){
  row <- rep(0,n_vars)
  for(i in paramedics) row[col_index(i,d)] <- 1
  A <- append(A,list(row)); dir <- c(dir,">="); rhs <- c(rhs, req$min_paramedics[d])
}

# Max 5 days/week
for(i in 1:n_staff){
  row <- rep(0,n_vars)
  for(d in 1:n_days) row[col_index(i,d)] <- 1
  A <- append(A,list(row)); dir <- c(dir,"<="); rhs <- c(rhs, 5)
}

# No >4 consecutive
for(i in 1:n_staff){
  for(start in 1:3){
    row <- rep(0,n_vars)
    for(d in start:(start+4)) row[col_index(i,d)] <- 1
```

```
    A <- append(A,list(row)); dir <- c(dir,"<="); rhs <- c(rhs,4)
  }
}

# Availability
unavailability <- list(
  c("P1",7),
  c("P4",1), c("P4",2),
  c("E2",6), c("E2",7),
  c("E5",5)
)
id_map <- setNames(1:n_staff, staff_ids)

for(u in unavailability){
  i <- id_map[[u[1]]]
  d <- as.numeric(u[2])
  row <- rep(0,n_vars); row[col_index(i,d)] <- 1
  A <- append(A,list(row)); dir <- c(dir,"=="); rhs <- c(rhs,0)
}

A_mat <- do.call(rbind,A)

solution <- lp(
  direction="min",
  objective.in=c_vec,
  const.mat=A_mat,
  const.dir=dir,
  const.rhs=rhs,
  all.bin=TRUE
)

optimal_cost <- solution$objval
optimal_cost
```

```
## [1] 11300
```

```
x_opt <- matrix(solution$solution, nrow=n_staff, byrow=TRUE)
rownames(x_opt) <- staff_ids
colnames(x_opt) <- req$day

x_opt
```

```
##    Monday Tuesday Wednesday Thursday Friday Saturday Sunday
## P1      1       0         1        1      1        1      0
## P2      0       1         1        0      1        1      1
## P3      1       1         0        1      1        0      1
## P4      0       0         1        1      1        1      0
## E1      1       1         0        0      0        0      1
## E2      1       1         1        0      1        0      0
## E3      1       0         0        1      1        1      1
## E4      0       1         0        1      1        1      1
## E5      0       1         1        1      0        1      1
```

```
## E6      1       0         1        1      1        1      0
## E7      0       0         1        1      1        1      0
## E8      0       0         1        0      1        0      0
```

5. Final Schedule Output Table

```
library(knitr)
```

```
## Warning: package 'knitr' was built under R version 4.3.3
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# Convert 0/1 to   or -
display_schedule <- ifelse(x_opt == 1, " ", "-")

schedule_df <- as.data.frame(display_schedule)
schedule_df$`Total Days` <- rowSums(x_opt)

# Individual cost
individual_cost <- function(i){
  total <- 0
  for(d in 1:ncol(x_opt)){
    wage <- if(d %in% c(6,7)) staff$weekend_wage[i] else staff$weekday_wage[i]
    total <- total + wage * x_opt[i,d]
  }
  total
}

schedule_df$Cost <- sapply(1:nrow(x_opt), individual_cost)
schedule_df <- cbind(Staff = staff$staff_id, schedule_df)

# Daily totals
daily_total <- colSums(x_opt)
paramedic_idx <- which(staff$type=="Paramedic")
paramedic_daily <- colSums(x_opt[paramedic_idx,])

summary_df <- rbind(
  schedule_df,
```

```
  c("Daily Total", as.character(daily_total), "", ""),
  c("Paramedics", as.character(paramedic_daily), "", "")
)

kable(
  summary_df,
  align="c",
  caption="Final EMS Optimal Weekly Schedule"
)
```

Table 1: Final EMS Optimal Weekly Schedule

	Staff	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	Total Days	Cost
P1	P1		-					-	5	1300
P2	P2	-			-				5	1350
P3	P3			-			-		5	1300
P4	P4	-	-					-	4	1050
E1	E1			-	-	-	-		3	580
E2	E2				-		-	-	4	720
E3	E3		-	-					5	980
E4	E4	-		-					5	980
E5	E5	-				-			5	980
E6	E6		-					-	5	940
E7	E7	-	-					-	4	760
E8	E8	-	-		-		-	-	2	360
13	Daily Total	6	6	8	8	10	8	6		
14	Paramedics	2	2	3	3	4	3	2		

```
cat("**Total Weekly Cost: $", format(optimal_cost, big.mark=','), "**", sep="")
```

```
## **Total Weekly Cost: $11,300**
```