Dear Chief Resident,

We are the team in charge of developing the residency schedule generation you wanted.

Our project is finished, our schedules generated from the data you provided. We will first explain how our mathematical model accommodates the requirements and drawbacks you outlined. Then, we will discuss some relative merits and drawbacks of each schedule delivered. Finally, we will outline limitations of the model and make suggestions for changes in future iterations of this project. We hope you will find this accessible, but also comprehensive enough to get a feel for what's going on under the hood.

"Hard constraints" are rules that *must* be true of *any* schedule the program outputs. The hard requirements you requested were that all residents be given at least their minimum residency requirements; that residents not be on-call on days they cannot work due to class; that residents are not booked to hospital shifts where they are not needed; and that every resident is at or below the monthly and 4-day-rolling shift limits. The program will not create a schedule that does not honor these basic requirements. You can rest assured of all of those..

"Soft constraints" are ones where there is some more wiggle room. We will explain first what soft constraints we defined, then how the program handles trading off between them. Our first soft constraint is minimizing the hours above minimum residency requirements any resident should be slated to work. One naive approach would be to just add up these hours for every resident, and then try to minimize that total; however, this runs the risk of generating schedules where one second year resident has 900 hours (the max possible in 1 year) and another has only 600, which is inequitable to residents. Instead, we opted to minimize the maximum number of hours *any one* resident is given over their minimum. What this means is, the system wants to give *every*one 1 shift above their minimum requirement, before giving *any*one 2 shifts. (This minimizing variable is $e^{+}_{\max,\mathrm{tot}}$ in the mathematical model; we will typeset such variables here as `e_{max,tot}^+`).

Residents prefer not to work more than 30% of their total hours on weekends, so we used the same trick to make sure nobody would be given significantly more weekend hours than anyone else (`e_{max,wk}^+`). Residents prefer not to be given Fridays, so we did this a third time to make sure nobody would be given significantly more Friday shifts than anyone else (`e_{max,Fri}^+`). For federal holidays, we wanted to enforce that people would be given *exactly* one federal holiday, so we used this trick in both directions, to make sure nobody would be given significantly more *or fewer* federal holiday shifts than anyone else (`e_{max,hol}^+, e_{max,hol}^-`).

Vacation days took more forethought. We figured people who request fewer vacation days would be upset if they were not granted those few days, but people who request many vacation days wouldn't be quite as upset if some of them were denied, as long as they still got some. Our system thus weights giving residents their first week's worth of vacation days (summed up over all residents under `e_{r,vac,wk1}^-`) more than the second weeks' worth (`e_{r,vac,wk2}^-`), and the second weeks' worth more than the following weeks' (`e_{r,vac,wk3}^-`). It turned out our system was able to give almost all vacation days requested anyway, but had things not worked out that way, our program would have tried to give everyone 1 week's worth of vacation days before giving anyone more than that, and then give everyone who wanted it 2 week's before giving still more.

Finally, the program includes a soft constraint to minimize the cost of hiring the EOC for the year.

The constants `c_1, c_2, …, c_7` in the mathematical model and the code are weighting constants assigned to each one of these soft constraints. To make the schedule prioritize one over the other as written here, you need only find the correct `c_whatever` and increase its value.

Now for the pros and cons of the generated schedules. For each schedule, the first setting deemphasizes the soft constraint of the EOC's cost, preferring instead to minimize overtime in all its forms (general, weekend hours, Friday hours) for each resident. The second one treats it as very important, and schedules the EOC for as few shifts as possible, preferring to use residents everywhere else. The outputs are surprisingly similar in many important facets -- for example, every x-year resident is given the same number of shifts, plus or minus 1, as every other x-year resident, which is excellent from a fairness standpoint. We are especially proud that almost all vacation hours were granted in our system, with the sole exceptions being in the low manpower schedule 3, under the "avoid hiring EOCs unless needed" weighting. Even there, it is 1 vacation day not granted apiece for 2 residents.

Nevertheless, every model has its limitations. For one, we weighted the preferences of 2nd and 3rd year residents equally in terms of hour placements; one could make the argument that 3rd years should have priority in that regard. Moreover, our model does not differentially weight Baker's primary and backup shifts, even though in reality the responsibilities of these two shifts probably vary quite a bit, and perhaps the backup shift should be preferentially given to the less senior member of the duo each night. (It added significant complexity, so we figured that if push came to shove, the two residents could work this out between themselves the night of.) Finally, we did not discriminate between the value of vacation days asked for in a range of time, rather than once-off vacation days. If someone wanted to plan a trip to Cancun for a week, it's useless for our model to give them six of those seven days (although giving fewer vacation days doesn't seem like much of a problem for us!).

We think we've done good work here. We hope you feel the same way. Thank you for your time, and let us know if there's anything we can do in the future to improve our results.

Kind Regards,

Nate, Sidney, Andrew, and Sreya.

Data Dashboards + Summaries

It's impossible to get a good idea for how a schedule is constructed just by glancing at the schedule itself. Hence we have created some aides.

On the next several pages, you will find high-level visual dashboards revealing the results our linear programming model outputted under two different setting conditions for each set of residents and vacation schedules generated. For each schedule/weighting pair, you will find

- **number of nights worked** for each resident,
- **hours worked beyond minimum** per resident *class*[1],
- **vacation days requested** as well as **vacation days granted** for each resident, and
- **total cost of calling in the EOC** for that given set of schedule and settings.

We are particularly proud that our model **grants almost every vacation day requested in all six cases**, with the exceptions being isolated to the schedule/setting pair with the least amount of total residents available. There is also **no favoritism between same-year residents** -- every year resident is allotted the same number of nights, plus or minus 1 single shift.

We have also attached **textual summaries of the schedules overall** after the visual dashboards, in case you want to see the "raw numbers" that we were looking for in our analysis.

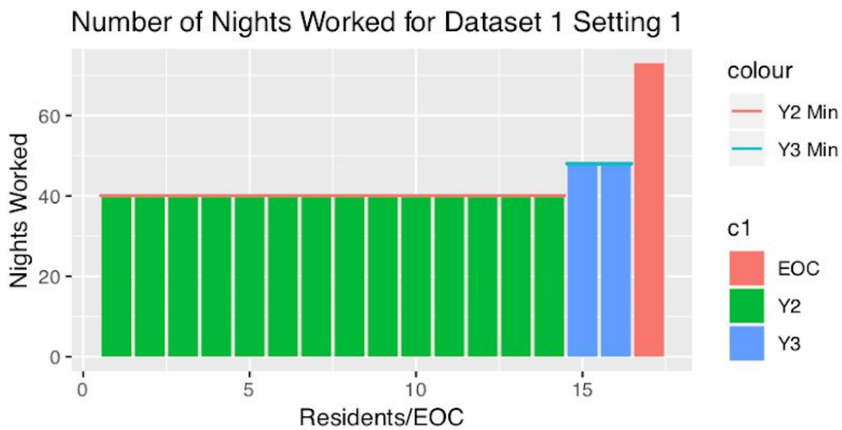
For example, if you wish to confirm that our program does indeed try to abstain from scheduling more than 30% of resident hours on weekends, this would be a good place to look.

We recommend you look at these while looking at the "Dashboard Variables" section of the Data-Independent Model if you wish to investigate these.

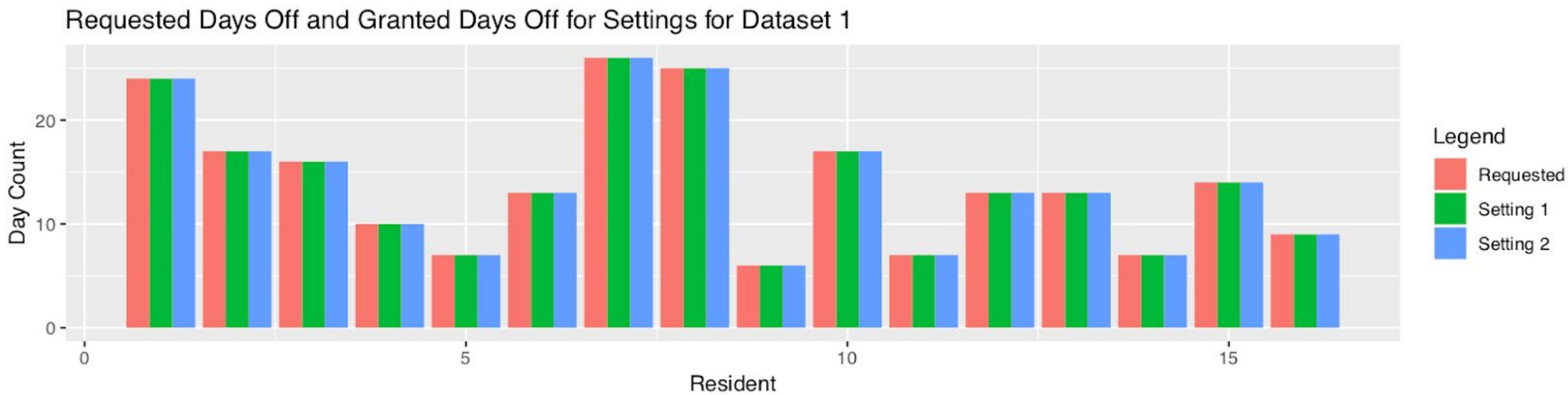
---

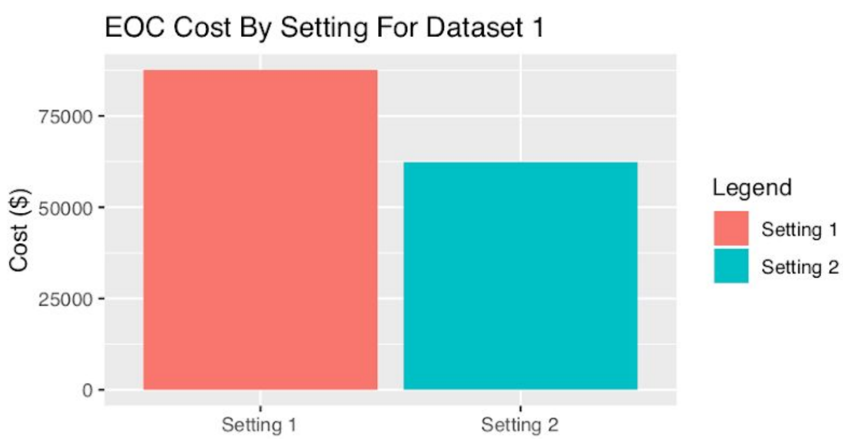
[1]Example: If 5 second-year residents worked one extra night beyond their 600 hour minimum, then this would appear as 5 * 15 = 75.

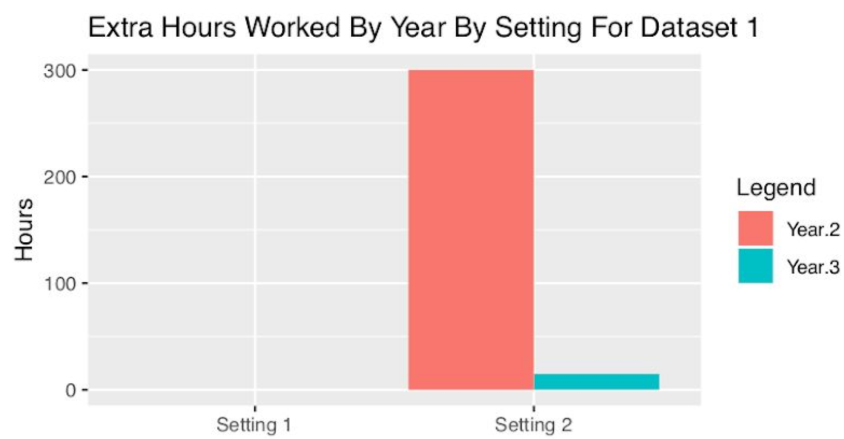
Our model does *not* favor one or the other residency class in terms of giving them hours beyond their minimum. The reason second years appear to work so many more hours on this part is simply because there are more of them than the third years.


Number of Nights Worked for Dataset 1 Setting 1
Nights Worked
0
20
40
60
0
5
10
15
Residents/EOC
colour
Y2 Min
Y3 Min
c1
EOC
Y2
Y3

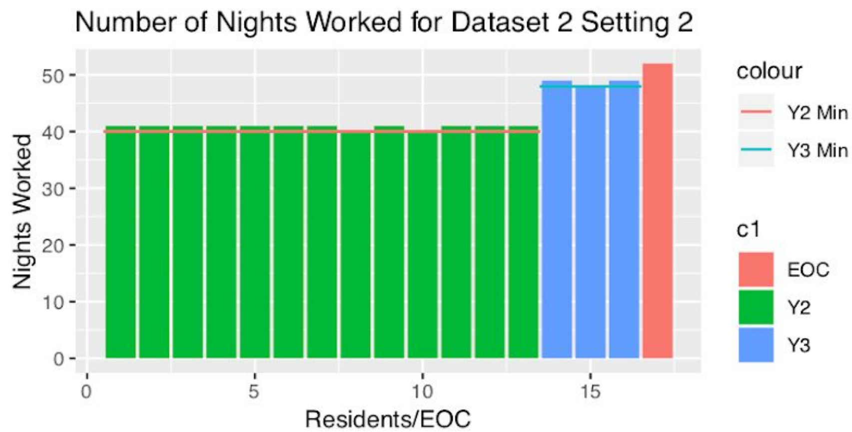

Number of Nights Worked for Dataset 1 Setting 2
Nights Worked
0
10
20
30
40
50
0
5
10
15
Residents/EOC
colour
Y2 Min
Y3 Min
c1
EOC
Y2
Y3

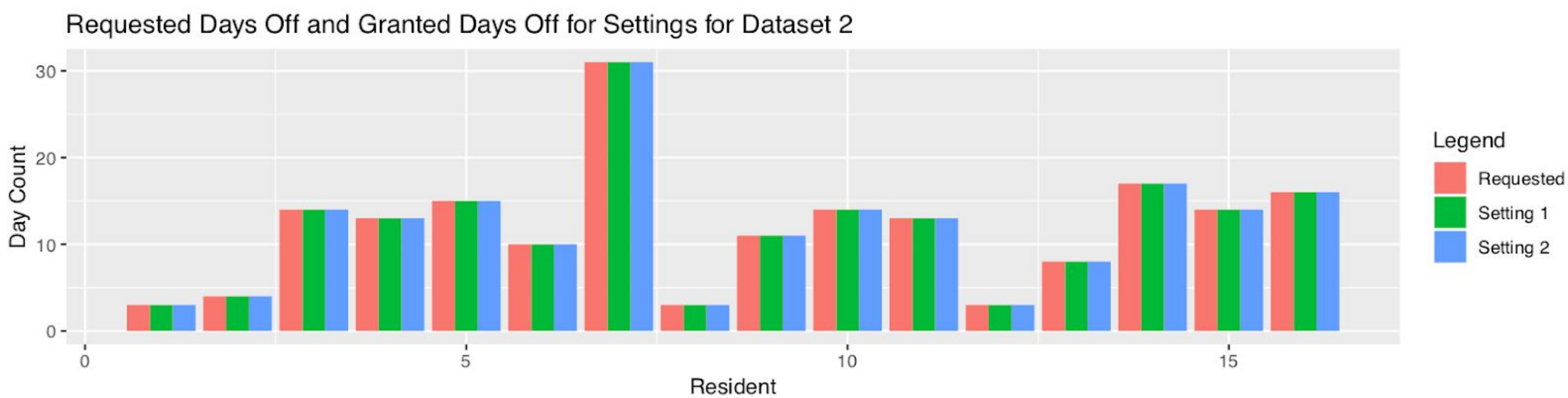

Requested Days Off and Granted Days Off for Settings for Dataset 1
Day Count
0
10
20
0
5
10
15
Resident
Legend
Requested
Setting 1
Setting 2

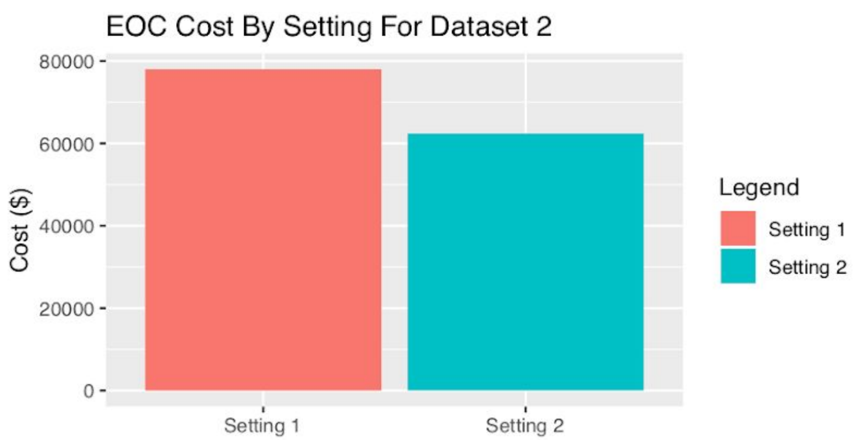

EOC Cost By Setting For Dataset 1
Cost ($)
0
25000
50000
75000
Setting 1
Setting 2
Legend
Setting 1
Setting 2

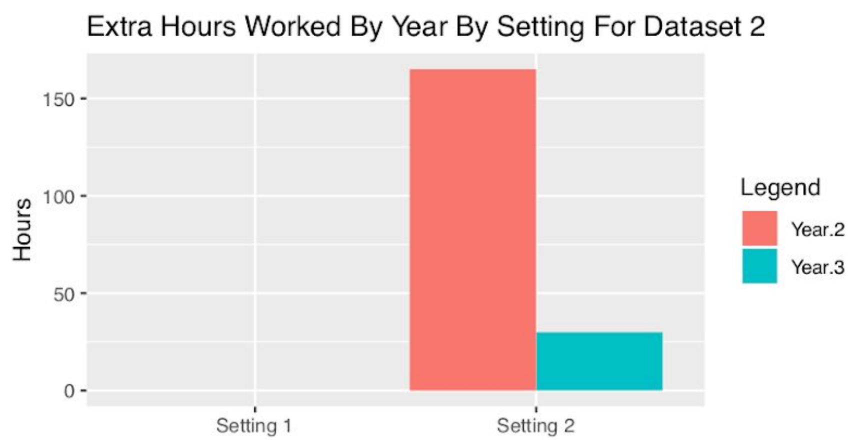

Extra Hours Worked By Year By Setting For Dataset 1
Hours
0
100
200
300
Setting 1
Setting 2
Legend
Year.2
Year.3

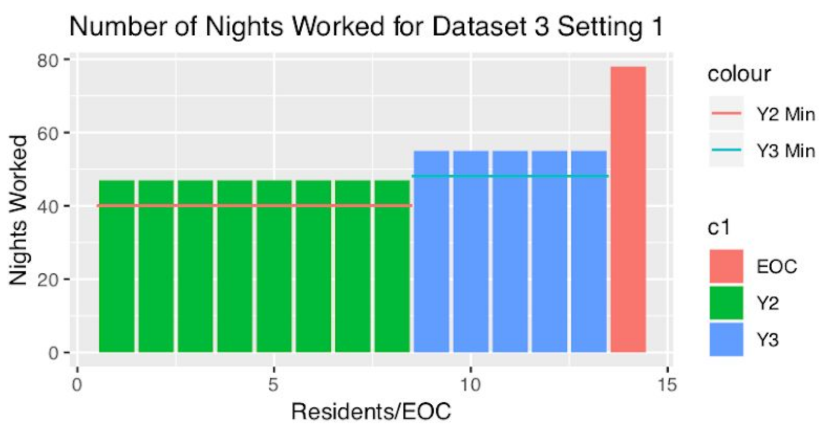

Number of Nights Worked for Dataset 2 Setting 1
Nights Worked
Residents/EOC
colour
Y2 Min
Y3 Min
c1
EOC
Y2
Y3


Number of Nights Worked for Dataset 2 Setting 2
Nights Worked
Residents/EOC
colour
Y2 Min
Y3 Min
c1
EOC
Y2
Y3

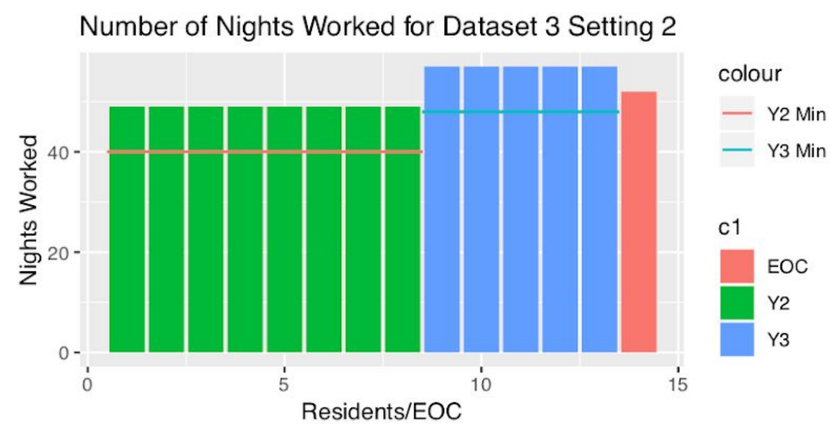

Requested Days Off and Granted Days Off for Settings for Dataset 2
Day Count
Resident
Legend
Requested
Setting 1
Setting 2


EOC Cost By Setting For Dataset 2
Cost ($)
Setting 1
Setting 2
Legend
Setting 1
Setting 2

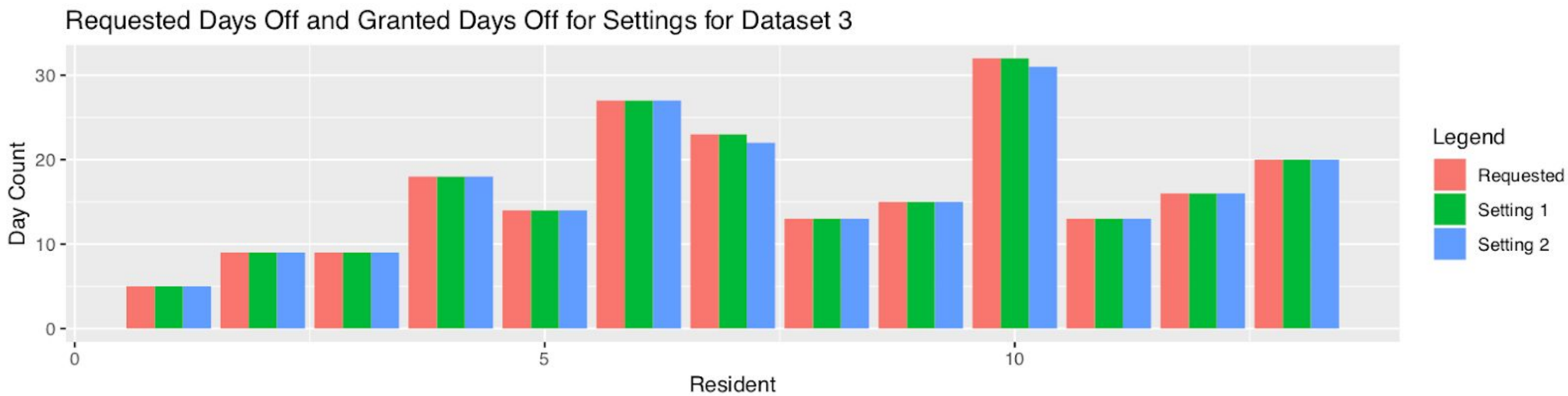

Extra Hours Worked By Year By Setting For Dataset 2
Hours
Setting 1
Setting 2
Legend
Year.2
Year.3



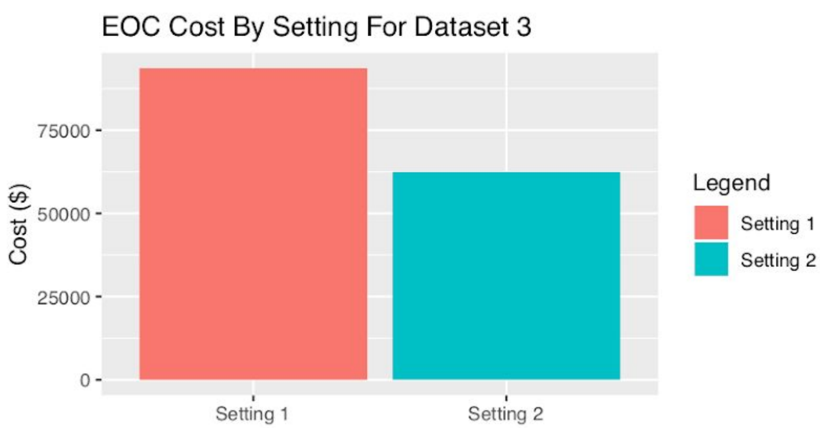
Number of Nights Worked for Dataset 3 Setting 1
Nights Worked
Residents/EOC
colour
Y2 Min
Y3 Min
c1
EOC
Y2
Y3



Number of Nights Worked for Dataset 3 Setting 2
Nights Worked
Residents/EOC
colour
Y2 Min
Y3 Min
c1
EOC
Y2
Y3



Requested Days Off and Granted Days Off for Settings for Dataset 3
Day Count
Resident
Legend
Requested
Setting 1
Setting 2



EOC Cost By Setting For Dataset 3
Cost ($)
Setting 1
Setting 2
Legend
Setting 1
Setting 2

Extra Hours Worked By Year By Setting For Dataset 3
Hours
Setting 1
Setting 2
Legend
Year.2
Year.3

Summary Statistics -- Data Dashboard 1, Schedule 1

```
# These are the summary statistics.

# t_r = Total number of hours assigned
# t_r_wk = Total number of weekend hours assigned
# t_r_Fri = Total number of Friday hours assigned
# t_r_vac = Total number of vacation hours assigned
# t_r_hol = Total number of federal holiday hours assigned

:      t_r  t_r_wk t_r_Fri t_r_vac t_r_hol    :=
1       600   180      90       0      15
2       600   180      90       0       0
3       600   180      90       0      15
4       600   180      90       0      30
5       600   180      90       0      15
6       600   165      90       0      30
7       600   180      90       0      15
8       600   180      90       0      30
9       600   180      90       0      30
10      600   180      90       0      15
11      600   180      90       0      15
12      600   180      90       0      15
13      600   180      90       0      15
14      600   180      90       0      15
15      720   210     105       0      30
16      720   210     105       0      30
EOC   1095      .       .       .       .
;

# Cost of EOCs for the year:

C_EOCs = 87600
```

Summary Statistics -- Data Dashboard 1, Schedule 2

```
# These are the summary statistics.

# t_r = Total number of hours assigned
# t_r_wk = Total number of weekend hours assigned
# t_r_Fri = Total number of Friday hours assigned
# t_r_vac = Total number of vacation hours assigned
# t_r_hol = Total number of federal holiday hours assigned

:      t_r t_r_wk t_r_Fri t_r_vac t_r_hol    :=
1      630   195     105       0      30
2      630   195      90       0      30
3      630   195      90       0      15
4      600   180      90       0      30
5      630   195      90       0      30
6      630   195     105       0      30
7      600   180      75       0      15
8      630   195     105       0      15
9      630   195     105       0      30
10     630   195     105       0      15
11     630   195     105       0      30
12     600   180      90       0      15
13     600   180      75       0      30
14     630   195     105       0      30
15     720   225     105       0      15
16     735   225     120       0      30
EOC    780     .       .       .       .
;

# Cost of EOCs for the year:

C_EOCs = 62400
```

Summary Statistics -- Data Dashboard 2, Schedule 1

```
# These are the summary statistics.

# t_r = Total number of hours assigned
# t_r_wk = Total number of weekend hours assigned
# t_r_Fri = Total number of Friday hours assigned
# t_r_vac = Total number of vacation hours assigned
# t_r_hol = Total number of federal holiday hours assigned

:      t_r t_r_wk t_r_Fri t_r_vac t_r_hol    :=
1      600   195      90      0      15
2      600   180      90      0      15
3      600   195      90      0      15
4      600   180      90      0      15
5      600   180      90      0      15
6      600   180      90      0      15
7      600   180      90      0      15
8      600   195      90      0      15
9      600   180      90      0      15
10     600   195      90      0      15
11     600   195      90      0      15
12     600   180      90      0      15
13     600   195      90      0      15
14     720   225     105      0      15
15     720   225     105      0      15
16     720   225     105      0      15
EOC    975     .       .       .       .
;

# Cost of EOCs for the year:

C_EOCs = 78000
```

Summary Statistics -- Data Dashboard 2, Schedule 2

```
# These are the summary statistics.

# t_r = Total number of hours assigned
# t_r_wk = Total number of weekend hours assigned
# t_r_Fri = Total number of Friday hours assigned
# t_r_vac = Total number of vacation hours assigned
# t_r_hol = Total number of federal holiday hours assigned

:      t_r t_r_wk t_r_Fri t_r_vac t_r_hol     :=
1      615   165     105       0       30
2      615   195     105       0       30
3      615   180      90       0       30
4      615   180      90       0       15
5      615   195     105       0       15
6      615   195      90       0       15
7      615   195      90       0       30
8      600   180      90       0       15
9      615   195      75       0       30
10     600   180      90       0       30
11     615   195      90       0       30
12     615   195      90       0       15
13     615   195     105       0       15
14     735   225     120       0       30
15     720   225     105       0       30
16     735   225     120       0       30
EOC    780     .       .       .        .
;

# Cost of EOCs for the year:

C_EOCs = 62400
```

Summary Statistics -- Data Dashboard 1, Schedule 2

```
# These are the summary statistics.

# t_r = Total number of hours assigned
# t_r_wk = Total number of weekend hours assigned
# t_r_Fri = Total number of Friday hours assigned
# t_r_vac = Total number of vacation hours assigned
# t_r_hol = Total number of federal holiday hours assigned

:      t_r  t_r_wk t_r_Fri t_r_vac t_r_hol    :=
1       705   210     105      0       30
2       705   210     105      0       30
3       705   210     105      0       30
4       705   210      90      0       30
5       705   210     105      0       30
6       705   210     105      0       15
7       705   210     105      0       15
8       705   210     105      0       15
9       825   240     120      0       30
10      825   240     120      0       15
11      825   240     120      0       30
12      825   240     120      0       30
13      825   240     120      0       30
EOC    1170     .       .        .       .
;

# Cost of EOCs for the year:

C_EOCs = 93600
```

Summary Statistics -- Data Dashboard 1, Schedule 2

```
# These are the summary statistics.

# t_r = Total number of hours assigned
# t_r_wk = Total number of weekend hours assigned
# t_r_Fri = Total number of Friday hours assigned
# t_r_vac = Total number of vacation hours assigned
# t_r_hol = Total number of federal holiday hours assigned

:      t_r t_r_wk t_r_Fri t_r_vac t_r_hol    :=
1      735   225     105       0      30
2      735   225     105       0      30
3      735   225     105       0      30
4      735   225     105       0      30
5      735   225     120       0      30
6      735   225     120       0      30
7      735   225     105      15      30
8      735   225     120       0      30
9      855   270     135       0      30
10     855   270     135      15      30
11     855   255     135       0      30
12     855   255     135       0      30
13     855   270     135       0      30
EOC    780     .       .       .       .
;

# Cost of EOCs for the year:

C_EOCs = 62400
```

IEMS 313 - Project 2, Data Independent Model

Sidney, Andrew, Nathaniel, Sreya

Last touched March 11, 2020

Sets and Indices

Quantifiers	Set	Description	Set relations
	$\mathcal{C}$	Set of calendar days (total)	
$\forall i \in \{1, 2, \ldots, 12\}$:	M_i	Set of days in month i from start	$M_i \subset \mathcal{C}$
$\forall i \in \{1, 2, \ldots, \lvert\mathcal{C}\rvert - 3\}$:	D_i	Set of 4-day rolling period: $\{i, i+1, i+2, i+3\}$	$D_i \subset \mathcal{C}$
	C_{hol}	Set of holidays	$C_h \subset \mathcal{C}$
	C_{wk}	Set of weekends	$C_{\text{wk}} \subset \mathcal{C}$
	C_{Fri}	Set of Fridays	$C_{\text{Fri}} \subset \mathcal{C}$
	$\mathcal{H}$	Set of hospital shifts	
$\forall h \in \mathcal{H}$:	C_h	Set of days hospital shift h needs coverage	$C_h \subset \mathcal{C}$
	$\mathcal{R}$	Set of all residents	
	$\mathcal{R}^\star$	Set of all residents *and* EOC	$\mathcal{R} \subsetneq \mathcal{R}^\star$
	R_2	Set of all second-year residents	$R_2 \cup R_3 = \mathcal{R}$
	R_3	Set of all third-year residents	$R_2 \cup R_3 = \mathcal{R}$
$\forall r \in \mathcal{R}^\star$:	C_r	Set of days resident or EOC r can work	$C_r \subset \mathcal{C}$
$\forall h \in \mathcal{H}, r \in \mathcal{R}^\star$:	$C_{h,r}$	Set of days hospital shift h can be covered by resident or EOC r	$C_{h,r} \triangleq C_h \cap C_r$
$\forall r \in \mathcal{R}$:	$\mathcal{V}_r$	Set of vacation days requested by resident r	$\mathcal{V}_r \subset \mathcal{C}$

Decision Variables

$\forall r \in \mathcal{R}^\star, h \in \mathcal{H}, c \in \mathcal{C}$:	$y_{r,h,c} \in \{0, 1\}$	Whether resident or EOC r is working hospital shift h, on calendar day c
$\forall r \in \mathcal{R}^\star, h \in \mathcal{H}, c \in \mathcal{C}$:	$x_{r,h,c}$	Hours worked by resident or EOC r, at hospital shift h, on calendar day c
$\forall r \in \mathcal{R}$:	$e^+_{r,\text{tot}}$	Hours worked by resident r, beyond minimum residency requirements
	$e^+_{\text{max,tot}}$	Highest number of overtime hours worked by any resident
$\forall r \in \mathcal{R}$:	$e^+_{r,\text{wk}}$	Hours worked by resident r on the weekends, beyond 30 pc. of their total
	$e^+_{\text{max,wk}}$	Highest number of overtime weekend hours worked by any resident
$\forall r \in \mathcal{R}$:	$e^+_{r,\text{Fri}}$	Hours worked by resident r on Fridays, beyond 15 pc. of their total
	$e^+_{\text{max,Fri}}$	Highest number of overtime Friday hours worked by any resident
	v_{max}	Highest total number of vacation hours requested by any resident
$\forall r \in \mathcal{R}$:	$e^-_{r,\text{vac}}$	Number of vacation hours resident r is given off
$\forall r \in \mathcal{R}$:	$e^-_{r,\text{vac,wk1}}$	Number of vacation hours resident r is given off for first vacation week
	$e^-_{\text{vac,wk1}}$	Sum of first-vacation-hour-week hours given to all residents
$\forall r \in \mathcal{R}$:	$e^-_{r,\text{vac,wk2}}$	Number of vacation hours resident r is given off for second vacation week
	$e^-_{\text{vac,wk2}}$	Sum of second-vacation-hour-week hours given to all residents
$\forall r \in \mathcal{R}$:	$e^-_{r,\text{vac,wk3}}$	Number of vacation hours resident r is given off for after second vacation week
	$e^-_{\text{vac,wk3}}$	Sum of post-second-vacation-hour-week hours given to all residents
	$e^-_{\text{min,vac}}$	Lowest number of vacation hours given off to any resident
$\forall r \in \mathcal{R}$:	$e^+_{r,\text{hol}}$	Holiday hours worked by resident r, above ideal
$\forall r \in \mathcal{R}$:	$e^-_{r,\text{hol}}$	Holiday hours worked by resident r, below ideal
	$e^+_{\text{max,hol}}$	Highest number of holiday hours above ideal assigned to any resident
	$e^-_{\text{max,hol}}$	Highest number of holiday hours below ideal assigned to any resident
$\forall r \in \mathcal{R}^\star$:	t_r	(“Dashboard variable”) Total hours for the year that each resident and EOC r assigned
$\forall r \in \mathcal{R}$:	$t_{r,\text{wk}}$	(“Dashboard variable”) Total weekend hours for the year that each resident r assigned
$\forall r \in \mathcal{R}$:	$t_{r,\text{Fri}}$	(“Dashboard variable”) Total Friday hours for the year that each resident r assigned
$\forall r \in \mathcal{R}$:	$t_{r,\text{vac}}$	(“Dashboard variable”) Total vacation hours for the year that each resident r assigned
$\forall r \in \mathcal{R}$:	$t_{r,\text{hol}}$	(“Dashboard variable”) Total holiday hours for the year that each resident r assigned
	$\mathbf{C}_{\text{EOCs}}$	(“Dashboard variable”) Total costs of hiring EOCs for the year.

Data

Quantifiers	Data	Description	Relation to other objects
$\forall i \in \{1, \ldots, 6\}$:	c_i	Weightings of error terms in objective function. **User-defined.**	
	α	Drop-off constant for objective function.	
	$\mathbf{r}_2$	Minimum residency hours for second-year students.	
	$\mathbf{r}_3$	Minimum residency hours for third-year students.	
	$\mathbf{d}_{\max}$	Maximum number of hours allowed for 4-day rolling period.	
	$\mathbf{m}_{\max}$	Maximum number of hours allowed per month.	
	$\mathbf{s}_{\max}$	Maximum number of hours a resident can work in one night.	
	$\mathbf{s}_h$	Hours a hospital shift needs to be filled for.	
	$\mathbf{p}_{\text{wk}}$	Maximum percent of hours residents prefer to fall on weekends.	
	$\mathbf{p}_{\text{Fri}}$	Maximum percent of hours residents prefer to fall on Fridays.	
	$\mathbf{e}_{\text{vac,cutoff}}$	Maximum hours to cap off vacation piecewise variables.	
	$\mathbf{h}_{\text{pref}}$	Preferred number of hours to schedule for federal holidays.	

Objective function

$$\min \quad c_1 e^+_{\text{max,tot}} + c_2 e^+_{\text{max,wk}} + c_3 e^+_{\text{max,Fri}} - c_4 e^-_{\text{min,vac}}$$

$$+c_5 \left(e^-_{\text{max,hol}} + e^-_{\text{max,hol}}\right) - c_6 \left[\alpha^{-1} e^-_{\text{vac,wk1}} + \alpha^{-2} e^-_{\text{vac,wk2}} + \alpha^{-3} e^-_{\text{vac,wk3}}\right]$$

This objective function seeks to minimize a weighted combination of:

- The maximum amount of overtime hours any resident is assigned.
- The maximum amount of weekend hours beyond preference any resident is assigned.
- The maximum amount of Friday hours beyond preference any resident is assigned.
- The maximum amount of holiday hours beyond preference any resident is assigned.
- The minimum amount of vacation hours granted to any resident.

$c_1, c_2, c_3, c_4, c_5, c_6 \geqslant 0$ are used for relative weightings at the discretion of the program-runner.

Constraints

Integer programming constraints

(On-call whole shift: If someone is on call for one hospital shift, they are on call for the entire shift.)

$$\forall r \in \mathcal{R}^\star, h \in \mathcal{H}, c \in \mathcal{C}\colon x_{r,h,c} = \mathbf{s}_{\max} \cdot y_{r,h,c}$$

Non-negativity constraints

(Non-negativity: No one can be registered for negative hours on any day. Total hours must also be non-negative, although this is not strictly needed.)

$$\forall r \in \mathcal{R}^\star, h \in \mathcal{H}, c \in \mathcal{C}\colon x_{r,h,c} \geqslant 0$$

(Non-negativity: None of our other resident-bound decision variables are allowed to be non-negative. You cannot work a negative number of “overtime”, “weekend overtime”, “Friday overtime”, holiday overshoot/undershoot, or vacation hours.)

$$\forall r \in \mathcal{R}\colon e^+_{r,\text{tot}}, e^+_{r,\text{wk}}, e^+_{r,\text{Fri}}, e^-_{r,\text{vac}}, e^-_{r,\text{vac,wk1}}, e^-_{r,\text{vac,wk2}}, e^-_{r,\text{vac,wk3}}, e^+_{r,\text{hol}}, e^-_{r,\text{hol}} \geqslant 0$$

(Non-negativity: Not strictly needed, as future constraints make all of these terms greater than or equal to other decision variables, but we include them here to be on the safe side. The highest number of “overtime”, "weekend overtime", "Friday overtime", or vacation hours either assigned to or requested by any resident must be non-negative.)

$$e^{+}_{\max,\text{tot}}, e^{+}_{\max,\text{wk}}, e^{+}_{\max,\text{Fri}}, \nu_{\max}, e^{-}_{\min,\text{vac}}, e^{+}_{\max,\text{hol}}, e^{-}_{\max,\text{hol}} \geqslant 0$$

Hard constraints

(No clones: A resident or EOC $r \in \mathcal{R}^\star$ can only work up to $\mathbf{s}_{\max}$ hours across all hospital shifts in one night.)

$$\forall r \in \mathcal{R}^\star \colon \forall c \in \mathcal{C} \colon \sum_{h \in \mathcal{H}} x_{r,h,c} \leqslant \mathbf{s}_{\max}$$

(Full staffing: Every hospital shift h needs every day in C_h fully covered, as in, for $\mathbf{s}_h$ hours exactly. No more, no less.)

$$\forall h \in \mathcal{H} \colon \forall c \in C_h \colon \sum_{r \in \mathcal{R}^\star} x_{r,h,c} = \mathbf{s}_h$$

(No scheduling for off days: A resident can't be scheduled to any day they can't work, or any day a given hospital shift isn't open.)

$$\forall r \in \mathcal{R}, h \in \mathcal{H} \colon \forall c \in \mathcal{C} \setminus C_{h,r} \colon x_{r,h,c} = 0$$

(Monthly limit: No resident may work more than $\mathbf{m}_{\max}$ hours, across all hospital shifts $h \in \mathcal{H}$, in a month.)

$$\forall r \in \mathcal{R} \colon \forall i \in \{1, 2, \ldots, 12\} \colon \sum_{c \in M_i} \sum_{h \in \mathcal{H}} x_{r,h,c} \leqslant \mathbf{m}_{\max}$$

(Four day limit: No resident may work more than $\mathbf{d}_{\max}$ hours, across all hospital shifts $h \in \mathcal{H}$, in a 4-day period.)

$$\forall r \in \mathcal{R} \colon \forall i \in \{1, 2, \ldots, |\mathcal{C}| - 3\} \colon \sum_{c \in D_i} \sum_{h \in \mathcal{H}} x_{r,h,c} \leqslant \mathbf{d}_{\max}$$

(Minimum residencies: Second-year residents must work at least $\mathbf{r}_2$ hours total, third years $\mathbf{r}_3$. Anything above that is extra and goes into their $e^{+}_{r,\text{tot}}$. *Change from Project 1: This was originally under "Soft constraints" because of the slack variable, but we were informed it would be a better fit here.*)

$$\forall r \in R_2 \colon \sum_{c \in \mathcal{C}} \sum_{h \in \mathcal{H}} x_{r,h,c} - e^{+}_{r,\text{tot}} = \mathbf{r}_2$$

$$\forall r \in R_3 \colon \sum_{c \in \mathcal{C}} \sum_{h \in \mathcal{H}} x_{r,h,c} - e^{+}_{r,\text{tot}} = \mathbf{r}_3$$

(Highest error term: This constraint ensures $e^{+}_{\max,\text{tot}}$ captures the highest amount of total "overtime" any resident is scheduled for.)

$$\forall r \in \mathcal{R} \colon e^{+}_{\max,\text{tot}} \geqslant e^{+}_{r,\text{tot}}$$

Soft constraints

(Weekend preferences: Residents prefer not to have more than $\mathbf{p}_{\text{wk}}$ (a percentage quantity) of their hours scheduled on the weekend. Anything above that goes into their $e^{+}_{r,\text{wk}}$. *Change from Project 1: We changed this to a $\leqslant$ expression, in order to avoid infeasibility if the system gives fewer than $\mathbf{p}_{wk}$ percent weekend hours.*

$$\forall r \in \mathcal{R} \colon \left(\sum_{c \in C_{\text{wk}}} \sum_{h \in \mathcal{H}} x_{r,h,c} \right) - e^{+}_{r,\text{wk}} \leqslant \mathbf{p}_{\text{wk}} \cdot \sum_{c \in \mathcal{C}} \sum_{h \in \mathcal{H}} x_{r,h,c}$$

(Highest error term: This constraint ensures $e^+_{\text{max,wk}}$ captures the highest amount of total "weekend overtime" any resident is scheduled for.)

$$\forall r \in \mathcal{R}\colon e^+_{\text{max,wk}} \geqslant e^+_{r,\text{wk}}$$

(Friday preferences: Residents prefer not to be scheduled for Friday, either. We decided to weight it so that residents prefer not to have have more than $\mathbf{p}_{\text{Fri}}$ (a percentage quantity) of their Friday hours scheduled. Anything above that goes into their $e^+_{r,\text{Fri}}$.) *Change from Project 1: We changed this to a $\leqslant$ expression, in order to avoid infeasibility if the system gives fewer than $\mathbf{p}_{Fri}$ percent Friday hours.*

$$\forall r \in \mathcal{R}\colon \left(\sum_{c \in C_{\text{Fri}}} \sum_{h \in \mathcal{H}} x_{r,h,c} \right) - e^+_{r,\text{Fri}} \leqslant \mathbf{p}_{\text{Fri}} \cdot \sum_{c \in \mathcal{C}} \sum_{h \in \mathcal{H}} x_{r,h,c}$$

(Highest error term: This constraint ensures $e^+_{\text{max,Fri}}$ captures the highest amount of total "Friday overtime" any resident is scheduled for.)

$$\forall r \in \mathcal{R}\colon e^+_{\text{max,Fri}} \geqslant e^+_{r,\text{Fri}}$$

(Vacation preferences: Residents prefer not to have their vacation hours denied. We compensate for this in two ways; here, we use $e^-_{r,\text{vac}}$ represents the number of vacation hours afforded to each resident.)

$$\forall r \in \mathcal{R}\colon \sum_{c \in \mathcal{V}_r} \left[\mathbf{s}_{\text{max}} - \left(\sum_{h \in \mathcal{H}} x_{r,h,c} \right) \right] = e^-_{r,\text{vac}}$$

(Lowest error term: This constraint ensures $e^-_{\text{min,vac}}$ captures the lowest amount of total vacation hours any resident is given off.)

$$\forall r \in \mathcal{R}\colon -e^-_{\text{min,vac}} \geqslant -e^-_{r,\text{vac}}$$

(Vacation piecewise: The variables $e^-_{r,\text{vac,wk1}}$, $e^-_{r,\text{vac,wk2}}$, and $e^-_{r,\text{vac,wk3}}$ are used to implement a piecewise linear function in the objective function, so that giving a resident more days off becomes less and less rewarded as they accrue more and more hours. The largest minimization term comes from $e^-_{r,\text{vac,wk1}}$, which stores up to $\mathbf{e}_{\text{vac,cutoff}}$ hours of given vacation in it to represent the first vacation week's worth of hours. $e^-_{r,\text{vac,wk1}}$ is less useful to add to in order to minimize the objective function, and is also capped at $\mathbf{e}_{\text{vac,cutoff}}$ hours to represent the second vacation week's worth. $e^-_{r,\text{vac,wk3}}$ has no cap, and is used to ensure additional hours awarded to residents past the third week are least attractive to the program.)

$$\forall r \in \mathcal{R}\colon e^-_{r,\text{vac,wk1}} \leqslant \mathbf{e}_{\text{vac,cutoff}}$$

$$\forall r \in \mathcal{R}\colon e^-_{r,\text{vac,wk2}} \leqslant \mathbf{e}_{\text{vac,cutoff}}$$

$$\forall r \in \mathcal{R}\colon e^-_{r,\text{vac,wk1}} + e^-_{r,\text{vac,wk2}} + e^-_{r,\text{vac,wk3}} = e^-_{r,\text{vac}}$$

(Vacation summation: We need to sum these terms up in order for them to have the correct overall effect on the objective function.)

$$e^-_{\text{vac,wk1}} = \sum_{r \in \mathcal{R}} e^-_{r,\text{vac,wk1}}$$

$$e^-_{\text{vac,wk2}} = \sum_{r \in \mathcal{R}} e^-_{r,\text{vac,wk2}}$$

$$e^-_{\text{vac,wk3}} = \sum_{r \in \mathcal{R}} e^-_{r,\text{vac,wk3}}$$

(Holiday hour preferences: Residents prefer to have as close to $\mathbf{h}_{\text{pref}}$ holiday hours as possible. We use $e^{+}_{r,\text{hol}}$ and $e^{-}_{r,\text{hol}}$ to tally up how much they either over- or under-shoot this target.)

$$\forall r \in \mathcal{R}\colon \left(\sum_{c \in C_{\text{hol}}} \sum_{h \in \mathcal{H}} x_{r,h,c} \right) - (e^{+}_{r,\text{hol}} - e^{-}_{r,\text{hol}}) = \mathbf{h}_{\text{pref}}$$

(Highest error term: These constraints ensure $e^{+}_{\text{max,hol}}$ and $e^{-}_{\text{max,hol}}$ respectively capture the highest amount of overshoot above and below the 15 holiday hours that any resident is scheduled for.)

$$\forall r \in \mathcal{R}\colon e^{+}_{\text{max,hol}} \geqslant e^{+}_{r,\text{hol}}$$

$$\forall r \in \mathcal{R}\colon e^{-}_{\text{max,hol}} \geqslant e^{-}_{r,\text{hol}}$$

Dashboard variable constraints

"Dashboard variables" are decision variables that are not core to the underlying logic of the linear program. We did not use them in writing the previous parts, because it is easier to double-check that the summations have been written correctly if they are written out in full. However, they do provide tidy sums and figures that are of use when trying to understand what the program is doing at a glance.

Since every such variable is defined as some sum of $x_{r,h,c}$ terms, which are all nonnegative, possibly scaled by a positive number, they are all themselves nonnegative by default.

(Total hours: The variable t_r reports the total hours for the year that each resident and EOC is assigned.)

$$\forall r \in \mathcal{R}^{\star}\colon t_r \triangleq \sum_{h \in \mathcal{H}} \sum_{c \in \mathcal{C}} x_{r,h,c}$$

(Total weekend hours: The variable $t_{r,\text{wk}}$ reports the total hours for the year that each resident is assigned for the weekend.)

$$\forall r \in \mathcal{R}\colon t_{r,\text{wk}} \triangleq \sum_{h \in \mathcal{H}} \sum_{c \in \mathcal{C}_{\text{wk}}} x_{r,h,c}$$

(Total Friday hours: The variable $t_{r,\text{Fri}}$ reports the total hours for the year that each resident is assigned for Fridays.)

$$\forall r \in \mathcal{R}\colon t_{r,\text{Fri}} \triangleq \sum_{h \in \mathcal{H}} \sum_{c \in \mathcal{C}_{\text{Fri}}} x_{r,h,c}$$

(Total vacation hours: The variable $t_{r,\text{vac}}$ reports the total hours for the year that each resident is assigned for their requested vacation days.)

$$\forall r \in \mathcal{R}\colon t_{r,\text{vac}} \triangleq \sum_{h \in \mathcal{H}} \sum_{c \in \mathcal{V}_r} x_{r,h,c}$$

(Total holiday hours: The variable $t_{r,\text{hol}}$ reports the total hours for the year that each resident is assigned for holidays.)

$$\forall r \in \mathcal{R}\colon t_{r,\text{hol}} \triangleq \sum_{h \in \mathcal{H}} \sum_{c \in C_h} x_{r,h,c}$$

(EOC cost: The variable $\mathbf{C}_{\text{EOCs}}$ reports the total cost of hiring EOCs for the year with the current schedule. EOCs are hired at \$80 per hour.)

$$\mathbf{C}_{\text{EOCs}} \triangleq 80 \cdot \sum_{r \in \mathcal{R}^{\star} \setminus \mathcal{R}} \sum_{h \in \mathcal{H}} \sum_{c \in C_h} x_{r,h,c}$$