Prescriptive Analytics in Health Care

Tutorial

International Summer School: Economic Modelling in Health Care

Riga, August 2018

Dr Sebastian Rachuba
Assistant Professor of Operations Management

University of Wuppertal
Schumpeter School of Business and Economics

rachuba@wiwi.uni-wuppertal.de

CAUTION

All models are wrong but some are useful!

George E.P. Box (1919 - 2013)

Tutorial

Part I

- Introduction to **R Studio** and **R**
- Develop your first optimisation model and solve it with R
- Formulate models for different planning problems

Part II

- Code models in R – potential to explore functions of the built-in solver of MS Excel
- Present findings
- Discuss benefits and drawbacks of optimisation

Tutorial

Part I – Example A

Imagine you run a laboratory that performs two tests (A and B). You have two assistants who run the tests. One the senior assistant works 10 hours a day, the junior assistant works only 8 hours per day. For each test you can generate a revenue of 10€. It takes the senior assistant 2 hours to finish test A and only one hour for test B. The junior assistant needs one hour to complete test A but requires 2 hours to complete test B.

1. What are the decisions you can make here?
2. What is your objective function?
3. What are the restrictions you need to take into account?
4. Formulate an optimisation model to maximise the revenue
5. What happens if
   1. The revenue was different?
   2. If you had a third test that gives you 15€ per test.
      Each of the assistants would need 3 hours to perform it.

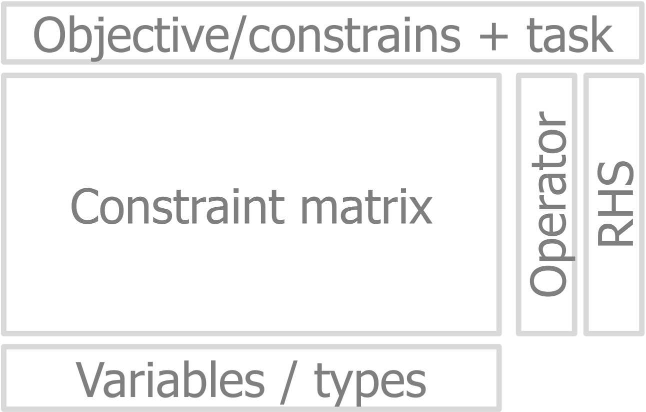
Solve optimisation models with R

- Standard form of optimisation model:

$$c_1 \cdot x_1 + c_2 \cdot x_2 + \ldots + c_n \cdot x_n \rightarrow \max$$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots a_{1n}x_n &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots a_{2n}x_n &\leq b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots a_{mn}x_n &\leq b_m \end{aligned}$$

$$x_1, x_2, \ldots, x_n \geq 0$$



Prescriptive Analytics

Optimisation in a nutshell

Left hand side (LHS): decision variables

Right hand side (RHS): parameters only

$$\sum_{a\in\mathcal{A}}\sum_{h\in\mathcal{H}} y^s_{ahj} = 1 \qquad \forall j \in \mathcal{J}, s \in \mathcal{S}$$

Operator, comparison

Repeat conditions

$$\max \sum_{s\in\mathcal{S}}\sum_{j\in\mathcal{J}}\sum_{a\in\mathcal{A}}\sum_{h\in\mathcal{H}} w^s \cdot y^s_{ahj} \cdot \ln(p_{ah} + \varepsilon)$$

$$\begin{aligned}
\text{s.t.}\quad & \sum_{a\in\mathcal{A}} \eta_{ag} \sum_{h\in\mathcal{H}} y^s_{ahj} = x^s_{jg} && \forall j \in \mathcal{J}, g \in \mathcal{G}, s \in \mathcal{S}\\
& \sum_{a\in\mathcal{A}}\sum_{h\in\mathcal{H}} y^s_{ahj} = 1 && \forall j \in \mathcal{J}, s \in \mathcal{S}\\
& \sum_{j\in\mathcal{J}}\sum_{a\in\mathcal{A}}\sum_{h\in\mathcal{H}} y^s_{ahj} \cdot c_h = C && \forall s \in \mathcal{S}\\
& \sum_{j\in\mathcal{J}} x^s_{jg} = d^s_g && \forall g \in \mathcal{G}, s \in \mathcal{S}\\
& \sum_{g\in\mathcal{G}_k}\sum_{j\in\mathcal{J}\setminus\mathcal{J}_k} x^s_{jg} = 0 && \forall k \in \tilde{\mathcal{K}}, s \in \mathcal{S}\\
& \sum_{h\in\mathcal{H}} z_{hj} = 1 && \forall j \in \mathcal{J}\\
& y^s_{ahj} \leq z_{hj} && \forall a \in \mathcal{A}, h \in \mathcal{H}, j \in \mathcal{J}, s \in \mathcal{S}\\
& z_{hj} \in \{0, 1\} && \forall h \in \mathcal{H}, j \in \mathcal{J}\\
& x^s_{jg} \in \mathbb{N}_0 && \forall j \in \mathcal{J}, g \in \mathcal{G}, s \in \mathcal{S}\\
& y^s_{ahj} \in \{0, 1\} && \forall a \in \mathcal{A}, h \in \mathcal{H}, j \in \mathcal{J}, s \in \mathcal{S}
\end{aligned}$$

Solve optimisation models with R

- Standard form of optimisation model:

$$c_1 \cdot x_1 + c_2 \cdot x_2 + \ldots + c_n \cdot x_n \rightarrow \max$$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots a_{1n}x_n &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots a_{2n}x_n &\leq b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots a_{mn}x_n &\leq b_m \end{aligned}$$

$$x_1, x_2, \ldots, x_n \geq 0$$

Typical structure:

$$c^t x$$

$$Ax \leq b$$

$$x \geq 0$$

Solve optimisation models with R

- **Install** packages lpSolve (only once)
  - `install.packages(“lpSolve”)`
- **Load** packages (any time you want to solve)
  - `library(lpSolve)`

Solve optimisation models with R

- Provide parameters of **objective function**
  - `obj.fun <- c(...)`
- Provide parameters of **coefficient matrix** and **right hand side**
  - `constr <- matrix(c(...), ncol=, byrow=TRUE)`
  - `constr.dir <- c(...)`
  - `rhs <- c(...)`

Solve optimisation models with R

- Provide parameters of **objective function**
  - `obj.fun <- c(...)`

- Provide parameters of **coefficient matrix** and **right hand side**
  - `constr <- matrix(c(...), ncol= number, byrow=TRUE)`
  - `constr.dir <- c(...)`
    - `“>”, “>=“, “=“, “<=“, “<“`
  - `rhs <- c(...)`

- Provide parameters of **objective function**
  - `model <- lp(direction, obj.fun, constr, constr.dir, rhs)`

Solve optimisation models with R

- Run **model** and give results & objective value:
  - `model`
  - `model$solution`
  - `model$objective`

Solve optimisation models with R

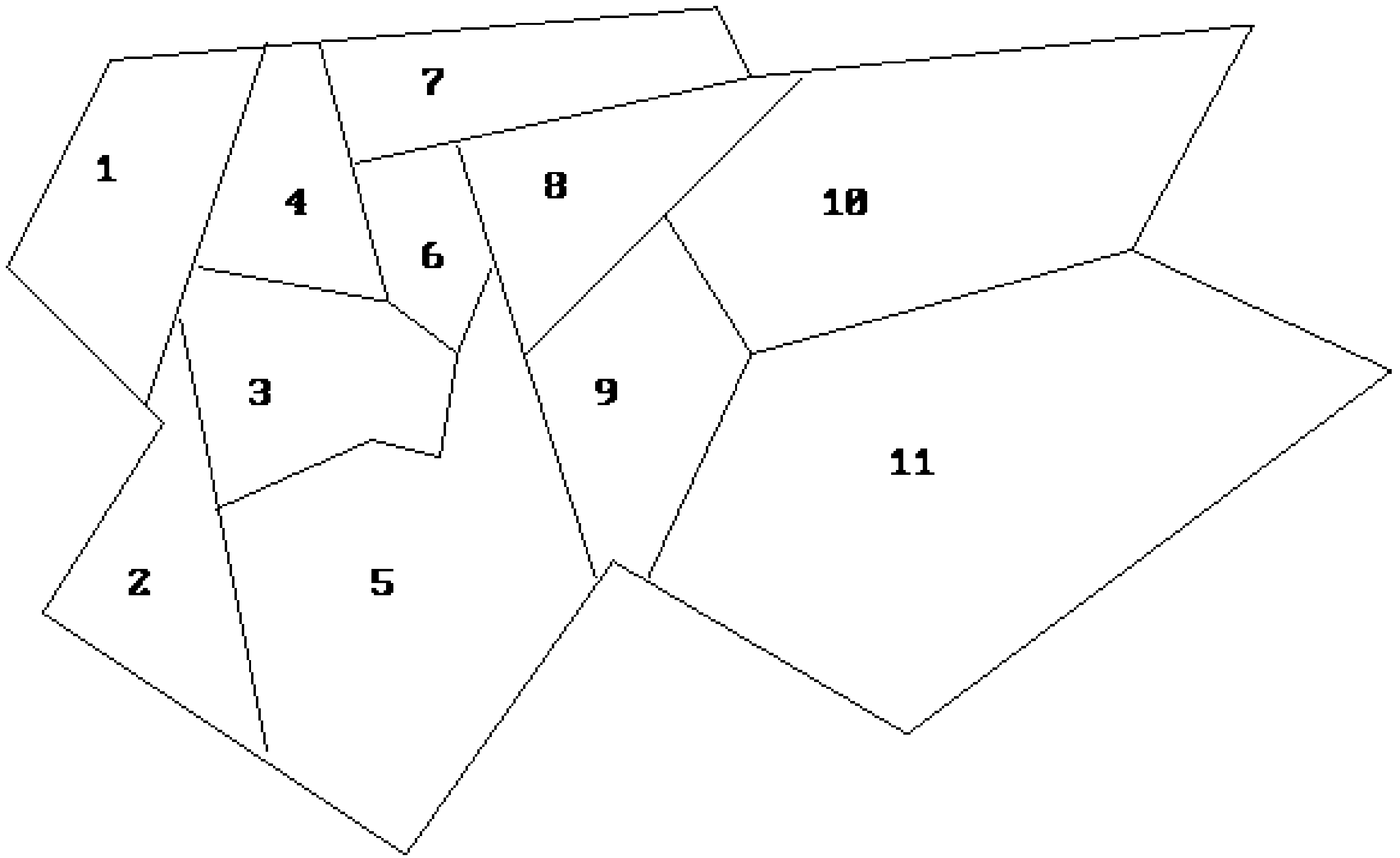
Solution for the model "Example A"

```
library(lp_model)
obj.fun <- c(10, 10)
constr <- matrix(c(2,1,1,2), ncol=2, byrow=TRUE)
constr.dir <- c("<=", "<=")
rhs <- c(10, 8)

model1 <- lp("max", obj.fun, constr, constr.dir, rhs)
model1
model1$solution
```

Tutorial

Part I – Example B



Source: http://mat.gsia.cmu.edu/classes/integer/node8.html

Tutorial

Part I – Locational problem

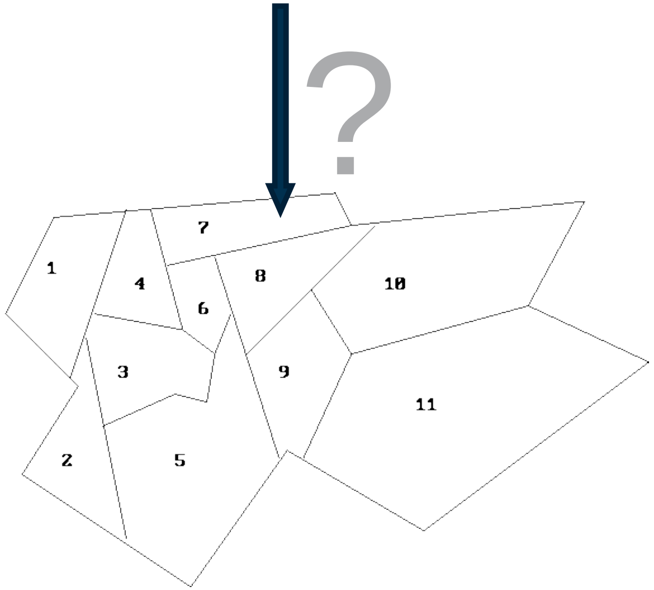
Tasks

1. Formulate an optimisation model which

   - Covers any demand region at least once

   - Uses a minimum number of locations

2. Consider that any location can be covered if department is placed in adjacent area
3. Formulate neighbourhood sets
4. Give explicit formulation of the optimisation model
5. Solve it
6. Present solutions



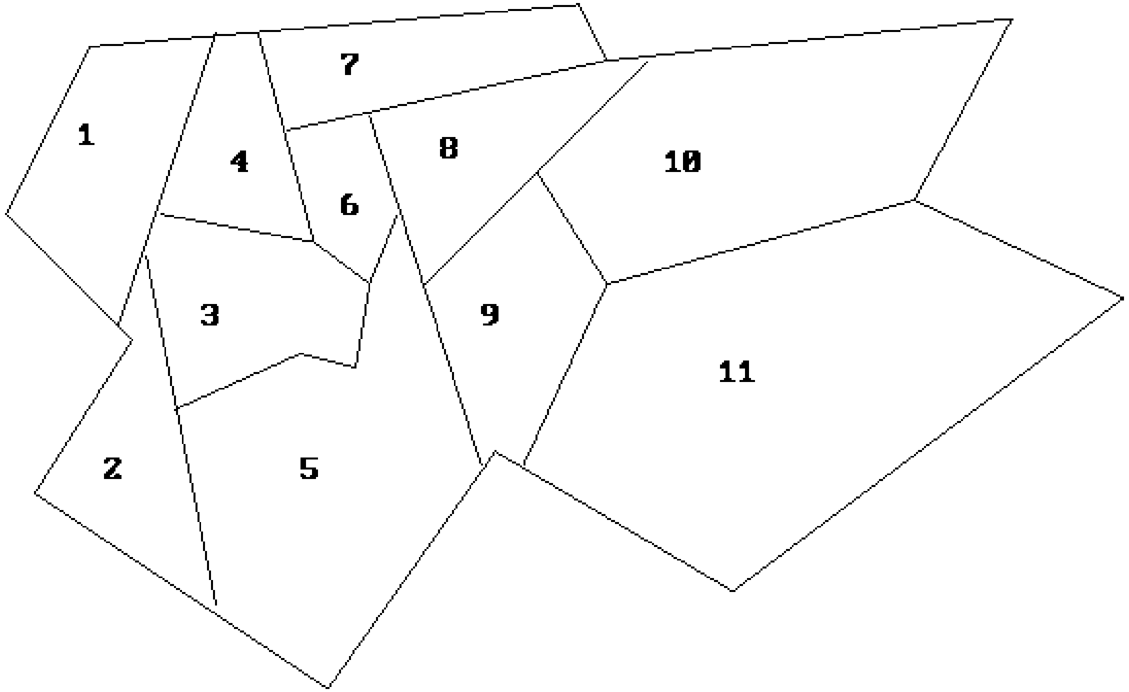
Source: http://mat.gsia.cmu.edu/classes/integer/node8.html

Tutorial

Part I – Locational problem

Neighbourhood sets

i	N_i	i	N_i
1	1,2,3,4	7	4,6,7,8
2	1,2,3,5	8	5,6,7,8,9,10
3	1,2,3,4,5,6	9	5,8,9,10,11
4	1,3,4,6,7	10	8,9,10,11
5	2,3,5,6,9	11	9,10,11
6	3,4,5,6,7,8,9		



Tutorial

Part I – Locational problem

Model formulation

$$
\begin{aligned}
\min \quad & x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} \\
\text{s.t.} \quad & x_1 + x_2 + x_3 + x_4 \geq 1 \\
& x_1 + x_2 + x_3 + x_5 \geq 1 \\
& x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \geq 1 \\
& x_1 + x_3 + x_4 + x_6 + x_7 \geq 1 \\
& x_2 + x_3 + x_5 + x_6 + x_8 + x_9 \geq 1 \\
& x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 \geq 1 \\
& x_4 + x_6 + x_7 + x_8 + x_9 \geq 1 \\
& x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} \geq 1 \\
& x_5 + x_8 + x_9 + x_{10} + x_{11} \geq 1 \\
& x_8 + x_9 + x_{10} + x_{11} \geq 1 \\
& x_j \in \{0,1\} \quad j = 1, 2, \ldots, 11
\end{aligned}
$$

Tutorial

Part I – Example C

Imagine you run the operating room at a small hospital. The hospital has one operating room which runs 8 hours a day. You have a very long waiting list and there are currently 20 patients that could be selected for surgery on the next day. The expected surgery times are given in the following table:

Patient	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Duration (in minutes)	35	17	48	23	121	230	47	87	134	55	29	90	65	40	76

1. What are the decisions you can make here?
2. What would be a suitable objective function?
3. What are the restrictions you need to take into account?
4. Formulate an optimisation model and solve it with R

Tutorial

Part I – Example D

The local hospital wants to determine the minimum number of administrative staff needed for its Emergency Department in order to deal with all the paperwork on different days of the week. Those staff members work 8 hours per day on five consecutive days. After five days of work they have to take two days off. The demand in working hours is: 48, 48, 52, 52, 36, 24, 24 (Mon – Sun).

1. What are the decisions you can make here?
2. What are possible "shift patterns"?
3. What is your objective function?
4. What are the restrictions you need to take into account?
5. Formulate an optimisation model to minimise the number of staff needed
6. What happens if …
   1. … the demand was different?
   2. … staff members would need to work 10 hour shifts and would have 2 days off.

Tutorial

Part II – Locational problem

Solution for the model "Example B"

```
constr <- matrix(c( 1,1,1,1,0,0,0,0,0,0,0,
                    1,1,1,0,1,0,0,0,0,0,0,
                    1,1,1,1,1,1,0,0,0,0,0,
                    1,0,1,1,0,1,1,0,0,0,0,
                    0,1,1,0,1,1,0,1,1,0,0,
                    0,0,1,1,1,1,1,1,1,0,0,
                    0,0,0,1,0,1,1,1,1,0,0,
                    0,0,0,0,1,1,1,1,1,1,0,
                    0,0,0,0,1,0,0,1,1,1,1,
                    0,0,0,0,0,0,0,1,1,1,1,
                    0,0,0,0,0,0,0,0,1,1,1), ncol=11, byrow=FALSE)
```

Tutorial

Part II – Locational problem

Solution for the model “Example B”

```
constr.dir <- c(rep(">=",11))
obj.fun <- c(1,1,1,1,1,1,1,1,1,1,1)
rhs <- c(rep(1,11))
model2 <- lp('min', obj.fun, constr, constr.dir, rhs, all.bin=TRUE)
model2
model2$solution
```

Solution for the model “Example C”

```
obj.fun <- c(rep(1,15))
constr <- matrix(c(
  35,17,48,23,121,230,47,87,134,55,29,90,65,40,76
), ncol=15, byrow=FALSE)
constr.dir <- c("<=")
rhs <- c(8*60)
or <- lp('max', obj.fun, constr, constr.dir, rhs, all.bin=TRUE)
or
or$solution
```

Tutorial

Part II – OR planning

Solution for the model "Example D"

```
obj.fun <- c(rep(1,7))
constr <- matrix(c( 8,0,0,8,8,8,8,
                    8,8,0,0,8,8,8,
                    8,8,8,0,0,8,8,
                    8,8,8,8,0,0,8,
                    8,8,8,8,8,0,0,
                    0,8,8,8,8,8,0,
                    0,0,8,8,8,8,8), ncol=7, byrow=FALSE)
constr.dir <- c(rep(">=",7))
rhs <- c(48,48,52,52,36,24,24)
shifts <- lp('min', obj.fun, constr, constr.dir, rhs, all.int=TRUE)
shifts
shifts$solutionon
```

Literature

Textbooks

- Ozcan, Yasar A (2017): Analytics and Decision Support in Health Care Operations Management, 3rd Edition, Wiley.
- Brandeau, Margaret L.; Sainfort, Francois; Pierskalla, William P. (Eds.): Operations Research and Health Care: A Handbook of Methods and Applications, Springer.
- Hall, Randolph (2013): Patient Flow – Reducing Delay in Healthcare Delivery, Springer.
- Vissers, Jan; Beech, Roger (2005): Health Operations Management: Patient Flow Logistics in Health Care, Routledge.

Research articles

- Salmon, A.; Rachuba, S.; Briscoe, S.; Pitt, M. (2018): A structured literature review of simulation modelling applied to Emergency Departments, in: Operations Research for Health Care (in press)
- Rachuba, S.; Ashton, L.; Knapp, K.; Pitt, M. (2018): Streamlining pathways for minor injuries in Emergency Departments through radiographer-led discharge, in: OR for Health Care (in press)
- Rachuba, S.; Salmon, A.; Zhelev, Z.; Pitt, M. (2018): Redesigning the diagnostic pathway for chest pain patients in emergency departments, in: Health Care Management Science, 21(2):177-191.
- Rachuba, S.; Werners, B. (2017): A fuzzy multi-criteria approach for robust operating room schedules, in: Annals of Operations Research, 251(1):325-350.
- Rachuba, S.; Werners, B. (2014): A robust approach for scheduling in hospitals using multiple objectives, in: Journal of the Operational Research Society, 65:546-556.

Useful links

- Script: Optimisation with R:
  http://www.is.uni-freiburg.de/resources/computational-economics/5_OptimizationR.pdf
- Using lpSolveAPI with R:
  https://www.r-bloggers.com/linear-programming-in-r-an-lpsolveapi-example/
- Set Covering problem:
  http://mat.gsia.cmu.edu/classes/integer/node8.html
- Knapsack problem:
  http://mat.gsia.cmu.edu/classes/integer/node6.html#SECTION00032000000000000000
- Modeling and Solving LPs with R (free book):
  https://www.r-bloggers.com/modeling-and-solving-linear-programming-with-r-free-book/
- Introduction to glpkAPI with R:
  https://cran.r-project.org/web/packages/glpkAPI/vignettes/glpk-gmpl-intro.pdf
- Introduction to linear programming:
  https://www.math.ucla.edu/~tom/LP.pdf

Contact details

Dr Sebastian Rachuba
Assistant Professor of Operations Management

University of Wuppertal
Schumpeter School of Business and Economics
Rainer-Gruenter-Str. 21, 42119 Wuppertal, Germany

Fon: +49 202 439-1437
Fax: +49 202 439-1107
E-Mail: rachuba@wiwi.uni-wuppertal.de

Solve optimisation models with R

- **Install** packages lpSolveAPI (only once)
  - `install.packages(“lpSolveAPI”)`
- **Load** packages (any time you want to solve)
  - `library(lpSolveAPI)`

Solve optimisation models with R

- Provide **data** matrix and number of variables
  - `d <- data.frame(x=c(…), y=c(…), w=c(…))`
  - `ncol <- 8`

- **Initiate** model and variable **types**
  - `lp_model <- make.lp(ncol=ncol)`
  - `set.type(lp_model, columns=1:ncol, type=c(“binary”))`

    - other types: integer

    - default: non-negative real

Solve optimisation models with R

- Define **objective** function with coefficients
  - `obj_coeff <- d[, “w”]`
  - `set.objfn(lp_model, obj_coeff)`
  - `lp.control(lp_rowpicker, sense=‘max’)` – other: ‘min’

Solve optimisation models with R

- Add constraints to model
  - **add.constraint(**

lp_model,	name of model
xt=c(1,1,1),	which rows, i.e. constraints
indices=c(1,2,3),	which variables of model (x_1, x_2, etc.)
rhs=1,	right hand side
type=“<=“	type of constraint
)	

- Repeat for more constraints

Solve optimisation models with R

- Solve model
  - `solve(lp_model)`
- Show output
  - `get.objective(lp_model)`
  - `get.variables(lp_model)`
- Print model
  - `lp_model`