Linear Programming Example

0.1 The problem

$$\max_{W,C} \quad 200\,W + 300\,C$$

$$\begin{aligned} &\text{subject to} \\ W + C &\le 500 \quad &&\text{(land)} \\ 3W + 4C &\le 1800 \quad &&\text{(labor)} \\ 4W + 3C &\le 2000 \quad &&\text{(fertilizer)} \\ W, C &\ge 0. \end{aligned}$$

0.1.1 Install + load package

```r
# install.packages("lpSolve")  # uncomment if not yet installed
library(lpSolve)
```

0.1.2 Set up and solve LP

```r
# Objective: max 200*xW + 300*xC
c <- c(200, 300)               # (xW, xC)

# Constraint matrix A (rows = constraints; cols = variables in same order as c)
A <- rbind(
  c(1, 1),    # land
  c(3, 4),    # labor
  c(4, 3)     # fertilizer
)

# Direction of each constraint and RHS vector b
dir <- c("<=", "<=", "<=")
b   <- c(500, 1800, 2000)

# Solve LP
sol <- lp(direction = "max",
          objective.in = c,
          const.mat = A,
          const.dir = dir,
          const.rhs = b,
          compute.sens = TRUE)   # also compute duals/sensitivity

stopifnot(sol$status == 0)  # 0 = optimal
```

0.2 Ranging Analysis Tables from `lpSolve` Output

```
# ---- Names for readability ----
var_names <- c("Wheat (x_W)", "Corn (x_C)")
con_names <- c("Land", "Labor", "Fertilizer")

# ---- Pull pieces from `sol` ----
obj_coef   <- sol$objective                 # c(200, 300)
x_opt      <- sol$solution                  # e.g., c(0, 450)
red_cost   <- tail(sol$duals, length(obj_coef))   # reduced costs for variables
coef_from  <- sol$sens.coef.from            # lower bounds for obj coefs
coef_to    <- sol$sens.coef.to              # upper bounds for obj coefs

# Helper to show Inf/large bounds nicely
fmt_inf <- function(x) ifelse(is.infinite(x) | abs(x) > 1e20, Inf, x)

# ---- Variable-ranging table (objective coefficient ranges) ----
var_tbl <- data.frame(
  variable            = var_names,
  objective_coef      = obj_coef,
  final_value         = x_opt,
  reduced_cost        = red_cost,
  allowable_increase  = fmt_inf(coef_to  - obj_coef),
  allowable_decrease  = fmt_inf(obj_coef - coef_from)
)

# ---- Constraint info ----
# For RHS ranges and shadow prices, `lpSolve` gives:
# - first `m` entries of `sol$duals` = shadow prices for constraints
# - `sol$duals.from` / `sol$duals.to` = absolute RHS bounds where that dual remains valid (binding ro
m <- length(con_names)
shadow     <- sol$duals[seq_len(m)]
rhs_from   <- sol$duals.from[seq_len(m)]
rhs_to     <- sol$duals.to[seq_len(m)]

# Reconstruct A and b to compute LHS and slack (clear & explicit for teaching)

lhs   <- as.vector(A %*% x_opt)
slack <- b - lhs

# Allowable ranges logic:
# - If constraint is binding (shadow>0 & slack 0), lpSolve provides absolute RHS bounds:
#     allowable_increase = rhs_to - RHS
#     allowable_decrease = RHS - rhs_from
# - If nonbinding (shadow 0), typical classroom interpretation:
#     allowable_decrease = slack (you can shrink RHS until it hits current use)
#     allowable_increase = Inf (shadow=0 persists while others bind)
allow_inc <- numeric(m)
allow_dec <- numeric(m)
for (i in seq_len(m)) {
  if (shadow[i] > 0 && abs(slack[i]) < 1e-9) {
    allow_inc[i] <- fmt_inf(rhs_to[i] - b[i])
    allow_dec[i] <- fmt_inf(b[i] - rhs_from[i])
```

```r
    } else {
      allow_inc[i] <- Inf
      allow_dec[i] <- slack[i]
    }
  }

con_tbl <- data.frame(
  constraint          = con_names,
  rhs                 = b,
  lhs_at_opt          = lhs,
  slack               = slack,
  shadow_price        = shadow,
  allowable_increase  = allow_inc,
  allowable_decrease  = allow_dec
)

# ---- Show tables ----
var_tbl
```

```
     variable objective_coef final_value reduced_cost allowable_increase
1 Wheat (x_W)            200           0          -25                 25
2  Corn (x_C)            300         450            0                Inf
  allowable_decrease
1                Inf
2           33.33333
```

```r
con_tbl
```

```
  constraint  rhs lhs_at_opt slack shadow_price allowable_increase
1       Land  500        450    50            0                Inf
2      Labor 1800       1800     0           75                200
3 Fertilizer 2000       1350   650            0                Inf
  allowable_decrease
1                 50
2               1800
3                650
```