Supply Chain Optimization Problem

Problem Statement

Multinational companies often have production facilities spread in countries around the world. While fulfilling the consumer demand, the companies need to consider various constraints pertaining to their supply chains - such as production costs (both fixed and variable), storage costs, freight costs, etc.

In order to make optimal decisions in view of these constraints, companies utilize various analytical methods. One such method commonly used for optimization is Linear Programming from the Operations Research area.

We have considered a Linear Programming Problem (LPP) for supply chain optimization in which a model has been created with the objective to minimize the total cost (comprising of fixed costs and variable costs) subject to various constraints.

Inputs

The supply chain consists of:

- Five markets: Brazil, USA, Japan, Germany and India
- Two types of manufacturing facilities: low-capacity site and high-capacity site

Key parameters include:

- Consumer demand for each country
- Manufacturing capacity by type of site
- CO_2 Emissions

The costs are comprised of:

- Fixed production costs
- Variable production costs
- Freight costs
- Storage costs

The tables below contain various inputs used for the model.

Consumer demand for each country

(Units/month)	Demand
USA	28,00,000
Germany	90,000
Japan	17,00,000
Brazil	1,45,000
India	1,60,000

Manufacturing capacity by type of site

Capacity (kUnits/month)	Low	High
USA	500	1000
Germany	500	1000
Japan	500	1000
Brazil	500	1000
India	500	1000

CO_2 Emissions

CO2 Emissions (kgs)	USA	Germany	Japan	Brazil	India
USA	0.000	84.062	299.317	125.522	181.069
Germany	84.062	0.000	267.654	140.791	149.308
Japan	299.317	267.654	0.000	282.625	127.726
Brazil	125.522	140.791	282.625	0.000	191.409
India	181.069	149.308	127.726	191.409	0.000

Fixed production costs

(k$/month)	Low	High
USA	6500	9500
Germany	4980	7270
Japan	6230	9100
Brazil	3230	4730
India	2110	6160

Variable production costs

($/Unit)	USA	Germany	Japan	Brazil	India
USA	12	12	12	12	12
Germany	13	13	13	13	13
Japan	10	10	10	10	10
Brazil	8	8	8	8	8
India	5	5	5	5	5

Freight costs

($/Container)	USA	Germany	Japan	Brazil	India
USA	0	12250	1100	16100	8778
Germany	13335	0	8617	20244	10073
Japan	15400	22750	0	43610	14350
Brazil	16450	22050	28000	0	29750
India	13650	15400	24500	29400	0

Storage costs

($/unit)	Low	High
USA	43.333	63.333
Germany	33.200	48.467
Japan	41.533	60.667
Brazil	21.533	31.533
India	14.067	41.067

Model

The model utilizes Linear Programming (LP), with the objective function aiming to minimize the total cost (comprising of fixed costs and variable costs) subject to various constraints. The solution from the model also helps to determine that in order to satisfy the consumer demand:

- which of the five countries the plants are to be located in (USA, Germany, Japan, Brazil and India), and
- which of the two types of plants are to be used (Low Capacity and High Capacity).

The modelling was carried out through the 'lpSolveAPI' library in R. The R code is included in Appendix 1, along with the objective function and the constraints.

There are 35 variables in the model:

- 25 variables indicating the number of units production by a country for a particular country (5 countries * 5 countries for which it could produce the units)

  The convention followed for the production variables is:
  sourcecountry_Production_targetCountry

  ```
  USA_Production_USA, Germany_Production_USA,
  Japan_Production_USA, Brazil_Production_USA,
  India_Production_USA, USA_Production_Germany,
  Germany_Production_Germany, Japan_Production_Germany,
  Brazil_Production_Germany, India_Production_Germany,
  USA_Production_Japan, Germany_Production_Japan,
  Japan_Production_Japan, Brazil_Production_Japan,
  India_Production_Japan, USA_Production_Brazil,
  ```

```
Germany_Production_Brazil, Japan_Production_Brazil,
Brazil_Production_Brazil, India_Production_Brazil,
USA_Production_India, Germany_Production_India,
Japan_Production_India, Brazil_Production_India,
India_Production_India
```

- 10 variables indicating the number of units in low capacity and high capacity (5 in each such that each of the 5 countries has one plan of each type - high and low)

  The convention followed for the location specific units is:
  country_Location_plantType
  (where plantType could be high capacity or low capacity plant)

```
USA_Location_Low, USA_Location_High, Germany_Location_Low,
Germany_Location_High, Japan_Location_Low,
Japan_Location_High, Brazil_Location_Low,
Brazil_Location_High, India_Location_Low,
India_Location_High
```

The objective function in the model is:

<u>Minimize</u>: Total costs = Sum of total fixed costs for low as well as high capacity plants across all the countries + Sum of variable costs associated with production made at and for each of the countries

The constraints in the model are:

- Number of units in high capacity plant + Number of units in low capacity plan in each country should be sufficient (>=) to accommodate the production by the country irrespective of the target country

```
Location constraint for USA: -500000 USA_Location_Low -
1000000 USA_Location_High +USA_Production_USA
+USA_Production_Germany +USA_Production_Japan
+USA_Production_Brazil +USA_Production_India <= 0;
Location constraint for Germany: -500000
Germany_Location_Low -1000000 Germany_Location_High
+Germany_Production_USA +Germany_Production_Germany
+Germany_Production_Japan +Germany_Production_Brazil
+Germany_Production_India <= 0;
Location constraint for Japan: -500000 Japan_Location_Low
-1000000 Japan_Location_High +Japan_Production_USA
+Japan_Production_Germany
+Japan_Production_Japan +Japan_Production_Brazil
+Japan_Production_India <= 0;
Location constraint for Brazil: -500000
Brazil_Location_Low -1000000 Brazil_Location_High
+Brazil_Production_USA +Brazil_Production_Germany
```

```
+Brazil_Production_Japan +Brazil_Production_Brazil
+Brazil_Production_India <= 0;
Location constraint for India: -500000 India_Location_Low
-1000000 India_Location_High +India_Production_USA
+India_Production_Germany
+India_Production_Japan +India_Production_Brazil
+India_Production_India <= 0;
```

- Production at a given target location country should be >= Demand for that location

```
Production constraint for USA: +USA_Production_USA
+Germany_Production_USA +Japan_Production_USA
+Brazil_Production_USA +India_Production_USA >= 2800000;
Production constraint for Germany: +USA_Production_Germany
+Germany_Production_Germany +Japan_Production_Germany
+Brazil_Production_Germany
+India_Production_Germany >= 90000;
Production constraint for Japan: +USA_Production_Japan
+Germany_Production_Japan +Japan_Production_Japan
+Brazil_Production_Japan +India_Production_Japan >=
1700000;
Production constraint for Brazil: +USA_Production_Brazil
+Germany_Production_Brazil +Japan_Production_Brazil
+Brazil_Production_Brazil
+India_Production_Brazil >= 145000;
Production constraint for India: +USA_Production_India
+Germany_Production_India +Japan_Production_India
+Brazil_Production_India +India_Production_India >=
160000;
```

- Total CO_2 emissions by the country while producing units for all the target countries put together should be less than the max allowed limit of 10000000000

```
CO2 constraint for USA: +84.0620948 Germany_Production_USA
+299.31716568 Japan_Production_USA +125.5220626
Brazil_Production_USA
+181.06915104 India_Production_USA <= 10000000000;
CO2 constraint for Germany: +84.0620948
USA_Production_Germany +267.65434 Japan_Production_Germany
+140.79148464 Brazil_Production_Germany
+149.30790664 India_Production_Germany <= 10000000000;
CO2 constraint for Japan: +299.31716568
USA_Production_Japan +267.65434 Germany_Production_Japan
+282.62497856 Brazil_Production_Japan
+127.72629136 India_Production_Japan <= 10000000000;
CO2 constraint for Brazil: +125.5220626
USA_Production_Brazil +140.79148464
Germany_Production_Brazil +282.62497856
Japan_Production_Brazil
+191.40908928 India_Production_Brazil <= 10000000000;
```

```
CO2 constraint for India: +181.06915104
USA_Production_India +149.30790664
Germany_Production_India +127.72629136
Japan_Production_India
+191.40908928 Brazil_Production_India <= 10000000000;
```

Results

Solving the LP model with the above constraints and the objective function, we obtained the following outcomes:

- **Number of possible solutions:** Model has only one optimal solution
- **Objective function:** Minimum optimized value for the total costs is 98109252
- **Model variable coefficients:**

```
USA_Location_Low            0
USA_Location_High           2.8
Germany_Location_Low        0
Germany_Location_High       0.09
Japan_Location_Low          0
Japan_Location_High         1.7
Brazil_Location_Low         0
Brazil_Location_High        0.145
India_Location_Low          0.320000000000001
India_Location_High         0
USA_Production_USA          2800000
Germany_Production_USA      0
Japan_Production_USA        0
Brazil_Production_USA       0
India_Production_USA        0
USA_Production_Germany      0
Germany_Production_Germany 90000
Japan_Production_Germany    0
Brazil_Production_Germany   0
India_Production_Germany    0
USA_Production_Japan        0
Germany_Production_Japan    0
Japan_Production_Japan      1700000
Brazil_Production_Japan     0
India_Production_Japan      0
USA_Production_Brazil       0
Germany_Production_Brazil   0
Japan_Production_Brazil     0
Brazil_Production_Brazil    145000
```

```
India_Production_Brazil    0
USA_Production_India       0
Germany_Production_India   0
Japan_Production_India     0
Brazil_Production_India    0
India_Production_India     160000
```

Model Validation (using LINGO)

We validated the model output from R using LINGO software, which gave the same results as our R code.

```
LINGO/WIN64 20.0.16 (19 Apr 2023), LINDO API 14.0.5099.259

 Global optimal solution found.
 Objective value:                              0.9810925E+08
 Infeasibilities:                               0.000000
 Total solver iterations:                              0
 Elapsed runtime seconds:                           0.11

Model Class:                                          LP

Total variables:                     35
 Nonlinear variables:                 0
 Integer variables:                   0

Total constraints:                   16
 Nonlinear constraints:               0

Total nonzeros:                     115
 Nonlinear nonzeros:                  0


                              Variable           Value        Reduced Cost
                      USA_LOCATION_LOW        0.000000            1761667.
                     USA_LOCATION_HIGH        2.800000            0.000000
                  GERMANY_LOCATION_LOW        0.000000            1353967.
                 GERMANY_LOCATION_HIGH       0.9000000E-01        0.000000
                    JAPAN_LOCATION_LOW        0.000000            1691200.
                   JAPAN_LOCATION_HIGH        1.700000            0.000000
                   BRAZIL_LOCATION_LOW        0.000000            870766.7
                  BRAZIL_LOCATION_HIGH       0.1450000            0.000000
                    INDIA_LOCATION_LOW       0.3200000            0.000000
                   INDIA_LOCATION_HIGH        0.000000            1952933.
                    USA_PRODUCTION_USA        2800000.            0.000000
                GERMANY_PRODUCTION_USA        0.000000            12.09013
                  JAPAN_PRODUCTION_USA        0.000000            12.99733
                 BRAZIL_PRODUCTION_USA        0.000000            7.648200
                  INDIA_PRODUCTION_USA        0.000000            1.334800
                USA_PRODUCTION_GERMANY        0.000000            13.49487
            GERMANY_PRODUCTION_GERMANY        90000.00            0.000000
              JAPAN_PRODUCTION_GERMANY        0.000000            21.59220
             BRAZIL_PRODUCTION_GERMANY        0.000000            14.49307
              INDIA_PRODUCTION_GERMANY        0.000000            4.329667
                  USA_PRODUCTION_JAPAN        0.000000            3.502667
              GERMANY_PRODUCTION_JAPAN        0.000000            9.774800
```

```
   JAPAN_PRODUCTION_JAPAN        1700000.            0.000000
  BRAZIL_PRODUCTION_JAPAN        0.000000            21.60087
   INDIA_PRODUCTION_JAPAN        0.000000            14.58747
    USA_PRODUCTION_BRAZIL        0.000000            24.90180
GERMANY_PRODUCTION_BRAZIL        0.000000            27.80093
  JAPAN_PRODUCTION_BRAZIL        0.000000            50.00913
 BRAZIL_PRODUCTION_BRAZIL        145000.0            0.000000
  INDIA_PRODUCTION_BRAZIL        0.000000            25.88660
     USA_PRODUCTION_INDIA        0.000000            21.09320
 GERMANY_PRODUCTION_INDIA        0.000000            21.14333
   JAPAN_PRODUCTION_INDIA        0.000000            24.26253
  BRAZIL_PRODUCTION_INDIA        0.000000            33.26340
   INDIA_PRODUCTION_INDIA        160000.0            0.000000

                    Row    Slack or Surplus      Dual Price
                      1        0.9810925E+08      -1.000000
                      2        0.000000            9.563333
                      3        0.000000            7.318467
                      4        0.000000            9.160667
                      5        0.000000            4.761533
                      6        0.000000            4.248133
                      7        0.000000           -21.56333
                      8        0.000000           -20.31847
                      9        0.000000           -19.16067
                     10        0.000000           -12.76153
                     11        0.000000           -9.248133
                     12        0.1000000E+11       0.000000
                     13        0.1000000E+11       0.000000
                     14        0.1000000E+11       0.000000
                     15        0.1000000E+11       0.000000
                     16        0.1000000E+11       0.000000
```

Further Scope

The model can potentially be expanded to:

- Consider the lead time and deadlines for delivering the units across locations.
- Consider the constraints on the number of containers that could be used to transport units.
- Impose restrictions on the container capacity.
- Associate the freight costs for the items that are shipped and not with the items that are produced as it would be more logical to do so.
- Analyze if it is possible to ship the items without producing them at a particular country, and if so, how to add the constraints for optimization.
- Identify the cost arbitrage and the overhead costs for producing the items at a particular location, and further fine tune the supply chain model.
- Analyze the solution obtained by Dual analysis using the same model (already included in our R code).

Appendix 1

R Code for Linear Programming Model

```
/* Objective function */
min: +6543333.33334 USA_Location_Low +9563333.33333 USA_Location_High
+5013200 Germany_Location_Low +7318466.66666 Germany_Location_High
+6271533.33334 Japan_Location_Low +9160666.66666 Japan_Location_High
+3251533.33333 Brazil_Location_Low
+4761533.33333 Brazil_Location_High +2124066.66667 India_Location_Low
+6201066.66666 India_Location_High
+12 USA_Production_USA +26.335 Germany_Production_USA +25.4
Japan_Production_USA +24.45 Brazil_Production_USA
+18.65 India_Production_USA +24.25 USA_Production_Germany +13
Germany_Production_Germany +32.75 Japan_Production_Germany
+30.05 Brazil_Production_Germany +20.4 India_Production_Germany +13.1
USA_Production_Japan +21.617 Germany_Production_Japan
+10 Japan_Production_Japan +36 Brazil_Production_Japan +29.5
India_Production_Japan +28.1 USA_Production_Brazil
+33.244 Germany_Production_Brazil +53.61 Japan_Production_Brazil +8
Brazil_Production_Brazil +34.4 India_Production_Brazil
+20.778 USA_Production_India +23.073 Germany_Production_India +24.35
Japan_Production_India +37.75 Brazil_Production_India
+5 India_Production_India;



/* Constraints */
Location constraint for USA: -500000 USA_Location_Low -1000000
USA_Location_High +USA_Production_USA +USA_Production_Germany
+USA_Production_Japan
+USA_Production_Brazil +USA_Production_India <= 0;
Location constraint for Germany: -500000 Germany_Location_Low -1000000
Germany_Location_High +Germany_Production_USA
+Germany_Production_Germany
+Germany_Production_Japan +Germany_Production_Brazil
+Germany_Production_India <= 0;
Location constraint for Japan: -500000 Japan_Location_Low -1000000
Japan_Location_High +Japan_Production_USA +Japan_Production_Germany
+Japan_Production_Japan +Japan_Production_Brazil +Japan_Production_India
<= 0;
Location constraint for Brazil: -500000 Brazil_Location_Low -1000000
Brazil_Location_High +Brazil_Production_USA +Brazil_Production_Germany
+Brazil_Production_Japan +Brazil_Production_Brazil
+Brazil_Production_India <= 0;
Location constraint for India: -500000 India_Location_Low -1000000
India_Location_High +India_Production_USA +India_Production_Germany
+India_Production_Japan +India_Production_Brazil +India_Production_India
<= 0;

Production constraint for USA: +USA_Production_USA
+Germany_Production_USA +Japan_Production_USA +Brazil_Production_USA
+India_Production_USA >= 2800000;
Production constraint for Germany: +USA_Production_Germany
```

```
+Germany_Production_Germany +Japan_Production_Germany
+Brazil_Production_Germany
+India_Production_Germany >= 90000;
Production constraint for Japan: +USA_Production_Japan
+Germany_Production_Japan +Japan_Production_Japan
+Brazil_Production_Japan +India_Production_Japan >= 1700000;
Production constraint for Brazil: +USA_Production_Brazil
+Germany_Production_Brazil +Japan_Production_Brazil
+Brazil_Production_Brazil
+India_Production_Brazil >= 145000;
Production constraint for India: +USA_Production_India
+Germany_Production_India +Japan_Production_India
+Brazil_Production_India +India_Production_India >= 160000;

CO2 constraint for USA: +84.0620948 Germany_Production_USA +299.31716568
Japan_Production_USA +125.5220626 Brazil_Production_USA
+181.06915104 India_Production_USA <= 10000000000;
CO2 constraint for Germany: +84.0620948 USA_Production_Germany
+267.65434 Japan_Production_Germany +140.79148464
Brazil_Production_Germany
+149.30790664 India_Production_Germany <= 10000000000;
CO2 constraint for Japan: +299.31716568 USA_Production_Japan +267.65434
Germany_Production_Japan +282.62497856 Brazil_Production_Japan
+127.72629136 India_Production_Japan <= 10000000000;
CO2 constraint for Brazil: +125.5220626 USA_Production_Brazil
+140.79148464 Germany_Production_Brazil +282.62497856
Japan_Production_Brazil
+191.40908928 India_Production_Brazil <= 10000000000;
CO2 constraint for India: +181.06915104 USA_Production_India
+149.30790664 Germany_Production_India +127.72629136
Japan_Production_India
+191.40908928 Brazil_Production_India <= 10000000000;
```

Code

```
# Supply chain analytics code
# set the working directory where the data files are present
setwd("E:\\IITK\\Optimization Methods for Analytics\\project\\data")

# install the necessary libraries and then include it
library(readr)
library(readxl)
#library(lpSolve)
library(lpSolveAPI)

# start reading the data files and convert those to dataframes. Remove
the unnecessary columns
variable_costs <- data.frame(read_excel("variable_costs.xlsx"))
rownames(variable_costs) <- variable_costs$Variable.Costs....Unit.
variable_costs <- variable_costs[, 2:6]
head(variable_costs)

freight_costs <- data.frame(read_excel("freight_costs.xlsx"))
rownames(freight_costs) <- freight_costs$Freight.Costs....Container.
freight_costs <- freight_costs[, 2:6]
head(freight_costs)
```

```
storage_costs <- data.frame(read_excel("storage_costs.xlsx"))
rownames(storage_costs) <- storage_costs$Storage.Costs....unit.
storage_costs <- storage_costs[, 2:3]
head(storage_costs)

fixed_costs <- data.frame(read_excel("fixed_costs.xlsx"))
rownames(fixed_costs) <- fixed_costs$...1
fixed_costs <- fixed_costs[, 2:3]
head(fixed_costs)

co2_emissions <- data.frame(read_excel("co2_emissions.xlsx"))
rownames(co2_emissions) <- co2_emissions$CO2.Emissions..kgs.
co2_emissions <- co2_emissions[, 2:6]
head(co2_emissions)

delivery_lead_times <-
  data.frame(read_excel("delivery_leadtime.xlsx"))
rownames(delivery_lead_times) <- delivery_lead_times$...1
delivery_lead_times <- delivery_lead_times[, 2:6]
head(delivery_lead_times)

capacity <- data.frame(read_excel("capacity.xlsx"))
rownames(capacity) <- capacity$Capacity..kUnits.month.
capacity <- capacity[, 2:3]
head(capacity)

demand <- data.frame(read_excel("demand.xlsx"))
rownames(demand) <- demand$X.Units.month.
#demand <- demand[,2:2]
head(demand)

# For further analysis - but missing data
delivery_deadlines <-
  data.frame(read_excel("delivery_deadlines.xlsx"))
rownames(delivery_deadlines) <- delivery_deadlines$...1
delivery_deadlines <- delivery_deadlines[, 2:6]
head(delivery_deadlines)

# Adding Freight costs to per unit variable costs since delivery details
are not available in the data
# Freight costs are for 1000 units
total_variable_costs <- variable_costs + freight_costs / 1000
head(total_variable_costs)

# Both fixed costs and storage costs are in 1000 $
total_fixed_costs <- (fixed_costs + storage_costs) * 1000
head(total_fixed_costs)

# limit on max CO2 emissions permitted by a country
max_co2_emission_permitted <- 10000000000

# Model development
# formulate LP

# create a model with x constraints, y variables
# Decision variables
# Fixed costs per location
```

```
# Variable costs for manufacturing it across locations
countries <- c("USA", "Germany", "Japan", "Brazil", "India")
lowhi <- c("Low", "High")
lst <- c("")

# total length = 2*length(countries)^2 + 2*length(countries)
vi <- 1

for (c in countries) {
  for (lh in lowhi) {
    lst[vi] <- paste (c, lh, sep = "_Location_")
    vi <- vi + 1
  }
}


for (c1 in countries) {
  for (c2 in countries) {
    lst[vi] <- paste (c2, c1, sep = "_Production_")
    vi <- vi + 1
  }
}


# Removing the delivery variables due to lack of data
#for(c1 in countries){
#  for(c2 in countries) {
#    lst[vi] <- paste (c1, c2, sep="_Delivery_")
#    vi <- vi + 1
#  }
#}

# set objective function
obj_coeffs <- rep(0, 35)
for (c in countries) {
  for (lh in lowhi) {
    obj_coeffs[which(lst == paste (c, lh, sep = "_Location_"))] <-
      total_fixed_costs[c, lh]
  }
  for (c1 in countries) {
    obj_coeffs[which(lst == paste (c, c1, sep = "_Production_"))] <-
      total_variable_costs[c, c1]
  }
}

# set constraints
clist = list()
slist <- c()
rlist <- c()
nlist <- c()
lconstraint <- 1

# add location fixed constraints
for (c in countries) {
  cc <- rep(0, 35)
  for (lh in lowhi) {
    cc[which(lst == paste (c, lh, sep = "_Location_"))] <-
      -1000 * capacity[c, lh]
```

```
  }
  for (c1 in countries) {
    cc[which(lst == paste (c, c1, sep = "_Production_"))] <- 1
  }
  clist[[lconstraint]] <- cc
  slist[lconstraint] <- "<="
  rlist[lconstraint] <- 0
  nlist[lconstraint] <- paste("Location constraint for", c, sep = " ")
  lconstraint <- lconstraint + 1
}

# add production constraints
for (c in countries) {
  cc <- rep(0, 35)
  for (c1 in countries) {
    cc[which(lst == paste (c1, c, sep = "_Production_"))] <- 1
  }
  clist[[lconstraint]] <- cc
  slist[lconstraint] <- ">="
  rlist[lconstraint] <- demand[c, "Demand"]
  nlist[lconstraint] <-
    paste("Production constraint for", c, sep = " ")
  lconstraint <- lconstraint + 1
}

# add co2 emission constraints
for (c in countries) {
  cc <- rep(0, 35)
  for (c1 in countries) {
    cc[which(lst == paste (c1, c, sep = "_Production_"))] <-
      co2_emissions[c1, c]
  }
  clist[[lconstraint]] <- cc
  slist[lconstraint] <- "<="
  rlist[lconstraint] <- max_co2_emission_permitted
  nlist[lconstraint] <- paste("CO2 constraint for", c, sep = " ")
  lconstraint <- lconstraint + 1
}

# add delivery lead time constraints
#cc <- rep(0,60)
#for(c in countries){
#  for(c1 in countries){
#    cc[which(lst == paste (c, c1, sep="_Dellivery_"))] <-
delivery_lead_times[c,c1]
#  }
#}
#clist[[lconstraint]] <- cc
#slist[lconstraint] <- "<="
#rlist[lconstraint] <- sum(delivery_deadlines)
#nlist[lconstraint] <- "Delivery lead time constraint"
#lconstraint <- lconstraint + 1


# develop the lp model
nconstraints <- length(rlist)
ndecisionvars <- length(lst)
```

```
lprec <- make.lp(nconstraints, ndecisionvars)

for (i in 1:ndecisionvars) {
  cc <- c()
  for (j in 1:nconstraints) {
    cc[j] <- clist[[j]][i]
  }
  set.column(lprec, i, cc)
}

# set objective function, constraints, names for constraints and columns
set.objfn(lprec, obj_coeffs)
set.constr.type(lprec, slist)
set.rhs(lprec, rlist)
dimnames(lprec) <- list(nlist, lst)

solve(lprec)
lprec


# write the final model to a file for easy browsing
write.lp(lprec, filename = "modelout.lp")

# identify the number of solutions
get.solutioncount(lprec)

# get the primal solution
get.primal.solution(lprec)

# dual solution
get.dual.solution(lprec)

# get the constraint coefficients
get.constraints(lprec)

# get the value of the objective function
get.objective(lprec)

# get the variable coefficients
print(cbind(lst,get.variables(lprec)))
```