Master in Statistics and Operative Research

Stochastic Programming

Benders' Decomposition of a Stochastic Problem

Víctor Diví i Cuesta

Universitat Politècnica de Catalunya

Facultat de Matemàtiques i Estadística

Contents

List of Codes

List of Tables

List of Summaries

1. Introduction

1.1. Problem

In the problem at hand, which we will call the ATM Deposit Problem, a bank branch wants to determine the amount of money to be deposited in an ATM on Fridays, before the weekend. The branch estimates that money has a cost (associated to the loss of benefits by interest rates) of c € for each € in the ATM. The demand of money during the weekend is a discrete random variable ξ, taking s values ξ_i with probabilities $p_i, i = 1, ..., s$. The ATM has a capacity of u €, with a technical minimum of l €. If the demand is greater than x, then the ATM has to be refilled, with a cost of q € for each € the demand exceeds x. The bank branch formulates the following stochastic optimization problem in extensive form:

$$\begin{aligned} \text{Min} \quad & cx + \sum_{i=1}^{s} p_i q y_i \\ \text{s.t.} \quad & l \leq x \leq u \\ & x + y_i \geq \xi_i \quad i = 1, ..., s \\ & y_i \geq 0 \quad i = 1, ..., s \end{aligned} \tag{P}$$

Fixing the value x to a valid value $\bar{x}$ ($l \leq \bar{x} \leq u$) and removing the now-constant cost from the objective function, from the extensive version of the problem (P) we obtain the following problem:

$$\begin{aligned} \text{Min} \quad & \sum_{i=1}^{s} p_i q y_i \\ \text{s.t.} \quad & y_i \geq \xi_i - \bar{x} \quad i = 1, ..., s \\ & y_i \geq 0 \quad i = 1, ..., s \end{aligned} \tag{Q}$$

and its dual, which will be the Bender's `Subproblem` used:

$$\begin{aligned} \text{Max} \quad & \sum_{i=1}^{s} u_i (\xi_i - \bar{x}) \\ \text{s.t.} \quad & u_i \leq p_i q \quad i = 1, ..., s \\ & u_i \geq 0 \quad i = 1, ..., s \end{aligned} \tag{$\mathsf{Q_D}$}$$

Using the `Subproblem` ($\mathsf{Q_D}$), we rewrite the original extensive problem (P) as follows:

$$\begin{aligned} \text{Min} \quad & cx + \max\left\{ \sum_{i=1}^{s} u_i (\xi_i - x) : 0 \leq u_i \leq p_i q, i = 1, ..., s \right\} \\ \text{s.t.} \quad & l \leq x \leq u \end{aligned} \tag{P'}$$

With the set of vertices $\{v^1, ..., v^V\}$ and extreme rays $\{r^1, ..., r^R\}$ of the feasible polyhedron of ($\mathsf{Q_D}$), we can rewrite (P') again as:

$$\begin{aligned} \text{Min} \quad & cx + z \\ \text{s.t.} \quad & z \geq \sum_{j=1}^{s} v_j^i (d_j - x) \quad i = 1, ..., V \\ & 0 \geq \sum_{j=1}^{s} r_j^i (d_j - x) \quad i = 1, ..., R \\ & l \leq x \leq u \end{aligned} \tag{BP}$$

For the Benders' Decomposition, we will use a relaxation (BPr) of (BP) in which we only use a subset of vertices and extreme points:

$$\begin{aligned} \text{Min} \quad & cx + z \\ \text{s.t.} \quad & z \geq \sum_{j=1}^{s} v_j^i (d_j - x) \qquad i \in \mathcal{V} \subseteq \{1, ..., V\} \\ & 0 \geq \sum_{j=1}^{s} r_j^i (d_j - x) \qquad i \in \mathcal{R} \subseteq \{1, ..., R\} \\ & l \leq x \leq u \end{aligned} \tag{BPr}$$

However, note that ($\mathsf{Q_D}$) is bounded, since all variables u_i have a lower bound (0) and an upper bound ($p_i q$). In fact, it is straightforward to see that the values of u_i will be 0 if $\xi_i - \bar{x} \leq 0$ and $p_i q$ otherwise. Therefore, its feasible polyhedron doesn't have any extreme ray, so we can simplify (BPr) to obtain our `Master Problem` formulation:

$$\begin{aligned} \text{Min} \quad & cx + z \\ \text{s.t.} \quad & z \geq \sum_{j=1}^{s} v_j^i (d_j - x) \qquad i \in \mathcal{V} \subseteq \{1, ..., V\} \\ & l \leq x \leq u \end{aligned} \tag{MP}$$

1.2. Scenario

The scenario used in this report have the following characteristics:

$$\begin{aligned} c &= 2.5 \cdot 10^{-4} \\ q &= 1.1 \cdot 10^{-3} \\ l &= 21 \cdot 10^{3} \\ u &= 147 \cdot 10^{3} \\ s &= 7 \end{aligned}$$

i	p_i	ξ_i
1	0.04	150
2	0.09	120
3	0.10	110
4	0.21	100
5	0.27	80
6	0.23	60
7	0.06	50

The units of l, u and ξ_i are thousands of €.
The scenario data in `AMPL` format is provided in the `atm.dat` file.

2. Extensive Version

The model used to solve the extensive version of the ATM Problem is shown in Code 1. The code is also provided in the `extensive.mod` file, along a small runner script in the `extensive.run` file.

```
# Parameters
set S;                      # Scenarios

param L >= 0;               # Minimum money to be deposited
param U > L;                # Maximum money to be deposited

param C;                    # Cost of deposited money €/€
param Q;                    # Cost of lack of money €/€

param P {S} >= 0 <= 1;      # Probability of each scenario
param D {S} >= 0;           # Demand of each scenario

# Variables
var x >= L <= U;            # Amount deposited
var y {S} >= 0;             # Missing amount in each scenario

# Model
minimize Total_Cost:
   C*x +
   sum {i in S} P[i] * Q * y[i];

subject to Demand {i in S}:
  x + y[i] >= D[i]

```

Code 1: AMPL Code for the Extensive model.

The results obtained with the execution of the extensive model can be seen in Summary 1. As can be seen, a total of 110k€ are deposited on the ATM, accounting for a cost of 27.5€. This amount falls short only in two scenarios, which adds 2.75€ to the expected cost for a total of 30.25€.

Cost	**Value**
Deposit	27.50
Shortage	2.75
Total	30.25

Variable	x	y_1	y_2	y_3	y_4	y_5	y_6
Value	110000	40000	10000	0	0	0	0

Summary 1: Results of the extensive model.

3. Benders' Decomposition

For the Benders' Decomposition, we use the formulations of the `Master Problem` and the `Subproblem` described in Section 1.1. The AMPL models for both are shown in Code 2, while the code for the actual Benders' Decomposition is shown in Code 3. The code is also provided in the `benders_simplified.mod` and `benders_simplified.run` files respectively.[1]

```
# Common Params
set S;                          # Scenarios

param P {S} ⩾ 0, <= 1;         # Probability of each scenario
param D {S} ⩾ 0;               # Demand of each scenario


# Master Problem
param NCuts ⩾ 0 integer;       # Number of cuts

param L ⩾ 0;                   # Minimum money to be deposited
param U > L;                    # Maximum money to be deposited

param C;                        # Cost of deposited money €/€

param Y {S, 1..NCuts};          # Missing amount in each scenario

var x ⩾ L, <= U;               # Amount deposited
var z;                          # Maximum cost for missing money

minimize Total_Cost: C * x + z;

subject to Cuts {k in 1..NCuts}:
   z ⩾ sum {i in S} Y[i,k] * (D[i] - x);


# Subproblem
param Q;                        # Cost of lack of money €/€
param X;                        # Amount deposited

var u {S} ⩾ 0;

maximize Dual_Cost: sum{i in S} u[i] * (D[i] - X);

subject to MissingCost {i in S}: u[i] <= P[i] * Q;
```

Code 2: AMPL Code for Benders' Decomposition models.

The AMPL code is a direct copy of the formulations presented above, with parameters in uppercase and variables in lowercase.

[1] Another pair of AMPL files (`benders.mod` and `benders.run`) are provided without the simplification of the `Master Problem`, i.e. using (BPr) instead of (MP). The only difference is that it handles both point and ray cuts instead of only point cuts. In practice, both are equivalent since ray cuts are never generated.

```
reset;

model benders.mod
data atm.dat;

option solver cplex;

# Problem declaration (variables, Objective_Function, Restrictions)
problem Master: x, z, Total_Cost, Cuts;
problem Sub: u, Dual_Cost, MissingCost;

# Initializations
let NCuts := 0;
let X := L;
let z := C * X;

param GAP default Infinity;
param epsilon default 1.0e-8;

repeat {
    solve Sub;
    display Sub.result, Dual_Cost;

    # Check gap
    let GAP := abs(Dual_Cost - z);
    display GAP;
    if GAP <= epsilon * z then break;

    # Add cut
    let NCuts := NCuts + 1;
    let {i in S} Y[i, NCuts] := u[i];

    solve Master;
    display Master.result, x, z, Total_Cost;

    let X := x;
}

display x, Total_Cost;
```

Code 3: AMPL Code for Benders' Decomposition.

We start the algorithm initializing $\bar{x}$. Then, at each iteration, we start solving the `Subproblem` (which is guaranteed to find a solution). Then, we check if the gap between the dual cost and z is lower than the accepted relative tolerance εz (with a fixed $\varepsilon = 10^{-8}$). If it is, we stop. Otherwise, we create an optimality cut with the values of $u_i, i = 1, ..., s$. We then proceed to solve the `Master Problem`, use the resulting x as the new value for $\bar{x}$, and move on to the next iteration.

Two scenarios have been executed, one in which we start with $\bar{x} = l$ and another in with $\bar{x} = u$. Both scenarios reach the same final results as the extensive model (shown in Summary 1). The evolution in the costs and $\bar{x}$ over the iterations for each scenario are shown in Tables 1 and 2. The Gap column shows the absolute gap between the dual cost of the same iteration and the value of z of the previous one.

As can be seen, both scenarios end with the same objective value as the extensive version in the same number of iterations. In fact, note that the costs between scenarios only differ in the first two iterations. In the second execution of the `Master Problem` (in iteration 2), we obtain the same result in both scenarios, and from that moment on, the results are always the same for both of them.

Iteration	**Dual Cost**	**Gap**	**Total Cost**	z	x
Init	-	-	-	5.25	21000
1	72.82	67.57	-29.03	-65.78	147000
2	0.132	65.912	24.0242	2.87833	84583.3
3	12.7417	9.86333	28.642	1.892	107000
4	3.509	1.617	30.2358	1.55158	114737
5	2.07263	0.521053	30.25	2.75	110000
6	2.75	3.55271e-15	-	-	-

Table 1: Evolution of costs for scenario l.

Iteration	**Dual Cost**	**Gap**	**Total Cost**	z	x
Init	-	-	-	36.75	147000
1	0.132	36.618	10.926	5.676	21000
2	72.82	67.144	24.0242	2.87833	84583.3
3	12.7417	9.86333	28.642	1.892	107000
4	3.509	1.617	30.2358	1.55158	114737
5	2.07263	0.521053	30.25	2.75	110000
6	2.75	3.55271e-15	-	-	-

Table 2: Evolution of costs for scenario u.

4. Conclusions

In this report we have solved an scenario of the stochastic ATM Deposit Problem using two different methods: solving directly the classical MILP formulation and using the Benders' Decomposition (with two different initial scenarios). We have presented a classic Benders' Master and subproblem formulations, as well as a simplification of the Master formulation that relies on the nature of the problem at hand. The computational results show that both methods (extensive version and Bender's Decomposition) yield exactly the same results.