NEW Take control of your code with Python control flow structures. You'll learn with real examples using loops, conditionals, try-except blocks, and pattern matching. May 28, 2025 intermediate python Module Six Notes "Linear Programming" Index to Module Six Notes 6.1: Problem Formulation 6.2: Computer Solution and Interpretation 6.3: Applications "It's interesting that our first winner was the inventor of the computer, and this year's winner (George Danzig) came up with one of the first business applications for the computer (Linear Programming)." Jeffrey Coors, President, Adolph Coors Company on the occasion of the presentation of the 1989 Coors American Ingenuity Award 6.1: Problem Formulation Introduction George Danzig is the father of the most powerful of all quantitative methods: linear programming, the subject of this and the final two modules of the course. He came up with his idea of developing a mathematical model to mechanize the military's planning process during World War II. After the war, once linear programming reached the private sector, it was quickly recognized as the most effective decision-support tool the world has ever seen. Linear programming's rise to prominence began in the oil companies in the 1950's. "We used linear programming to schedule our tanker fleets, design port facilities, blend gasoline, create financial models, you name it," says Bill Drew, former manager of research for Exxon. Quickly linear programming became commonly used to: - solve environmental problems - make decisions to improve the economy of small countries - help investors make the best portfolio decisions - schedule worldwide air travel - develop global business strategies - improve government functions and ideas - schedule automobile assembly line production My first application of linear programming came about at the Pentagon when I joined a joint service task force to create a unified transportation command. We envisioned that a unified command would have the command and control needed to enable the use of linear programming to develop optimum airlift deployment schedules to quickly move troops and their equipment into hot spots around the world. Although I retired before it's the first real test (Desert Storm), I was pleased to read the after action reports of the success of the deployment model. Models such as the one used by the unified transportation command are common place in distribution systems in the private sector. All linear programming problems have the following operational characteristics: - an objective function to optimize, such as: -- which product/service mix to select so as to maximize profit contribution (product mix problem) -- which product blend to select so as to minimize production cost (blending problem) -- which supply chain locations to select for a distribution system so as to minimize product throughput time and cost (transportation problem) -- which airline pilots to tap so as to minimize labor costs (scheduling problem) -- which capital budget projects to select so as to maximize net present value (capital budgeting problem) -- which stocks to place in a portfolio so as to maximize return (portfolio selection problem) - "subject to" constraints: -- labor, machine time, inventory, supplier (for the product mix problem) -- hops, barley and malt recipe (for a blending problem - at Coors Brewery for example) -- facility capacity (for the transportation problem) -- pilot availability and passenger/route demand (for the scheduling problem) -- project relationships and available capital (for the capital budgeting problem) -- investment funds available and risk tolerance (for the portfolio problem) and linear programming problems all have the following mathematical characteristics: - Decision alternatives are expressed as decision variables - A single linear objective function to maximize or minimize - Linear constraints that set upper limits on resources or lower bounds on requirements - Nonnegative real values (including fractions) for the decision variables The last three characteristics can be thought of as assumptions, since we have to assume that real world problems can be modeled as single objective problems, with linear objective and constraint equations, and fractions allowed as values for the decision variables. More on these assumptions as we get into extensions to linear programming to address these issues. Problem Formulation We begin solving linear programming problems with the problem. Let's start with a classic product mix problem - one of the earliest applications of linear programming. A company plans to make 3 models (A, B, and C) of their main product next month. Production capacity is limited to 100 total - each model takes the same about of production time. There is enough cycle stock inventory to make 200 models, except for the signature finishing paint. A supplier problem results in only 1,000 gallons being available next month. Each Model A requires 5 gallons of paint, each Model B requires 7 gallons and each Model C requires 10 gallons. Marketing wants the following mix: exactly 20 Model A's; at least 5 Model B's; and no more than 2 Model C's for every Model B produced. The firm wants to select that product mix so as to maximize profit contribution. Here are the numbers that go into profit contribution: Table 6.1.1 Model Revenue Cost Model A $16,000 $12,000 Model B $18,200 $13,000 Model C $20,000 $17,000 To formulate the linear programming problem means to translate the word problem statement into mathematical equations called the objective function and constraint set. The first step in the formulation is to name the decision variables and their units of measurement unless the units of measurement are obvious. The decision variables for this problem are: A = the number of units of Model A to produce next month B = the number of units of Model B to produce next month C = the number of units of Model C to produce next month The second step is to formulate the objective function. Here, management must be very careful to precisely articulate what is to be optimized. In Table 6.1.1, we see that we could decide a production schedule (how many A's, B's and C's to produce) so as to minimize costs. Alternatively, we could decide a production schedule so as to maximize revenue. Finally, we have sufficient data that we could decide a production schedule to maximize profit contribution = revenue - cost. Generally, when variable costs and revenues are known, the firm would like to determine that production schedule which maximizes profit contribution, so as not to sub-optimize with a production schedule. The objective function to maximize profit contribution is written: Maximize Z = (16,000 - 12,000)A + (18,200 - 13,000)B + (20,000 - 17,000)C Maximize Z = 4,000 A + 5,200 B + 3,000 C Z is the commonly selected symbol to represent the value of the objective function, in this case, profit contribution. We add the word "maximize" or "minimize" to keep our focus on the appropriate objective. Note first that there is only one objective. Second, this is a linear algebraic equation with each decision variable raised to the first power (no squared terms, for example). Third, the equation represents the additive property (no cross-products of the decision variables like B*C, no fractions involving the decision variables such as B/C or 1/B - only addition and subtraction are allowed in the equation). If this was the end of the formulation, what should the manufacturer do? That's right, build an infinite number of Model B's since Model B's have the highest profit contribution. But, we know there are at least resource constraints that restrict infinite profit contribution. For example, we know that production capacity is limited to 100 total units. What if the firm decided to build all Model B's - then they would build 100 Model B's. The profit contribution would be: Z = 4,000 *(0) + 5,200 * (100) + 3,000 * (0) = $ 520,000 That would be a great profit contribution but it would be an infeasible solution since we know that marketing wants to get a broader product line out next month (e.g., exactly 20 Model A's). The power of linear programming is that we can optimize the objective function while mathematically considering all of the organizational constraints. Let's now formulate the constraint set. The first constraint is production capacity being limited to 100 total units. this constraint is written: A + B + C < 100 This linear equation says all of the A's, plus all of the B's, plus all of the C's that are produced must be less than or equal to 100. Note that I made this constraint an inequality rather than a strict equality. When there is a choice, it is better to allow slack in a constraint as this provides maximum flexibility for other constraints. If 5 A's are built, 5 B's and 5 C's, then 15 total units are produced. There is slack of 85 units, or there are 85 units of unused production capacity. If 5 A's, 20 B's and 75 C's are built, then 100 total units are produced, and there is no slack. In this case, we say the constraint is binding on the solution - the firm can't make any more profit unless it gets some more production capacity. Another note to make is that all of the decision variables are on the left hand side of the equation, and the constant is on the right hand side. This is standard linear programming convention, and this is how the equation will be entered into The Management Scientist. The next constraint in the word problem concerns a general inventory constraint: A + B + C < 200 Note that this constraint will have slack as long as the production capacity constraint limits the production to 100 units. Thus, you might suggest leaving out this constraint. There is nothing wrong with that, except in really big production problems (e.g. 10 decision variables, 50 constraints), it is very difficult to keep track of cross-constraint relationships - it is easier to leave all of the constraints in and let the software identify the binding and non-binding constraints, including those that are redundant. Next comes the paint constraint: 5 A + 7 B + 10 C < 1,000 This constraint shows coefficients other than an implied "1" in front of the decision variables as in the first two constraints. These coefficients are called technological coefficients which represent the rate of usage of the limited resource. For example, if the firm makes 1 Model A, then 5 gallons of paint are taken from the 1,000 available. If 2 Model A's are built, then 10 gallons are taken from the 1,000 available. Another important note at this point is that this constraint has units of measurement in gallons, whereas the first two constraints had units of measurement in units of production. Units of measurement can be different across the constraints, but must be the same within a constraint. That is, the left hand side of the equation units of measurement must be the same as the right hand side units of measurement. In this case, note that the right hand side units of measurement are 1,000 gallons. The left hand side units of measurement are also gallons since units cancel out in the following coefficient expression: 5 gallons / unit * A units = 5 A gallons I don't want to make this seem complicated - it will come natural to you when you write constraint equations. The last three constraints concern the marketing product line mix constraints. First, marketing wants exactly 20 units of Model A to be built. That is a strict equality because of the exactly: A = 20 Next, marketing wants at least 5 B's to be built. This inequality is written: B > 5 If the firm builds 10 units of Model B, then there would be a surplus of 5 units. Finally, we have a ratio and proportion constraint. It is best to right this constraint exactly as the word problem presents the ratio and proportion: C is to B as 2 is to 1, or in equation form with the "no more than": C / B < 2 / 1 This constraint is legal mathematically, but is not a proper linear programming constraint because of the fraction C / B. So, multiple both sides of the equation by B, and move the decision variables to the left hand side: C < 2 B; and - 2 B + C < 0 Let's summarize the objective function and the constraints: Maximize Z = 4,000 A + 5,200 B + 3,000 C Subject to: 1) A + B + C < 100 2) A + B + C < 200 3) 5 A + 7 B + 10 C < 1,000 4) A = 20 5) B > 5 6) - 2 B + C < 0 That's it except for the implied constraints A > 0, B > 0 and C > 0 (remember then non-negativity condition I mentioned earlier). Linear programming computer software packages like The Management Scientist automatically include the non-negativity constraints so we don't have to bother about them when using software to solve these type of problems. My favorite part of linear programming is problem formulation - converting organizational considerations into mathematical expressions - but then I enjoy trying to organize the mob at the Publix Deli into a single line to minimize average time in line! After the problem is formulated, we proceed to its solution. Problem Solution We already talked about an infeasible solution - any solution that doesn't work in one or more of the constraints. For example, if we select the solution: A = 20, B = 20 and C = 60 let's look at the numbers: Z = 4,000 (20) + 5,200 (20) + 3,000 (60) = $364,000 1) 20 + 20 + 60 = 100 which is = 100: slack = 0; constraint is binding 2) 20 + 20 + 60 = 100 which is < 200: slack = 100 constraint is non binding 3) 5 (20) + 7 (20) + 10 (60) = 840 which is < 1,000: slack = 160; constraint is non binding 4) 20 = 20; constraint is binding 5) 20 > 5: surplus = 15 6) - 2 (20) + 60 = 20 which is not < 0, this solution is infeasible. If we built 20 B's, the most we could build of C's would be 40 (no more than 2 C's for every B). The next, and more interesting class of solutions are the infinite set of feasible solutions. We can come up with a feasible solution by inspection since this is a small problem. Looking at the constraint that was infeasible, let's simply limit the solution to 40 C's and run the numbers: A = 20, B = 20 and C = 40 Z = 4,000 (20) + 5,200 (20) + 3,000 (40) = $304,000 1) 20 + 20 + 40 = 80 which is < 100: slack = 20 2) 20 + 20 + 40 = 80 which is < 200: slack = 120 3) 5 (20) + 7 (20) + 10 (40) = 640 which is < 1,000: slack = 360 4) 20 = 20; constraint is binding 5) 20 > 5: surplus = 15 6) - 2 (20) + 40 = 0 which is < 0 (the concept of slack/ surplus would not apply to a ratio and proportion constraint) Since this solution results in a feasible solution to each constraint, the solution is feasible. Even for small problems, there are many feasible solutions. We could try a few more, always trying to increase profit contribution subject to meeting the constraints. Fortunately, there are algorithms which quickly determine the most interesting of these many solution: that solution which is the optimal feasible solution. The next section introduces The Management Scientist Linear Programming Module for solving these types of problems. Before we look at The Management Scientist Linear Programming Model, I want to introduce one more example. Both of these will be worked by the software in Module 6.2 Notes. A Medical Clinic Resource Allocation Model A small clinic specializes in general and orthopedic surgery. Each general surgery performed nets $200 to the clinic, and each orthopedic surgery performed nets $300. There are two constraints of interest: surgery and therapy. Surgery is limited to 80 person-hours per week, and therapy to 120 person-hours per week. Each general patient needs 2 hours of surgery time and 2 hours of therapy. Each orthopedic patient needs 4 hours of surgery and 12 hours of therapy. How many general and orthopedic patients should the clinic schedule to maximize its net profit? The decision variables for this problem will be: General = Number of General Surgery patients to schedule for surgery per week Orth = Number of Orthopedic patients per week The linear programming problem formulation is then: Maximize Z = 200 General + 300 Orth Subject to: 1) Surgery Time: 2 General + 4 Orth < 80 2) Therapy Time: 2 General + 12 Orth < 120 I selected this example as a small example so that we can easily find the optimal solution, and then be able to replicate this with the software package. This will help us understand the The Management Scientist linear programming output. Note that I abbreviated orthopedic to Orth since the software only allows 8 characters in the names of the decision variables. With a two decision variable, two constraint maximization problem, there are only three possible solutions to evaluate. One solution is to set Orth = 0 and see as many general patients as possible (the more patients, the higher the profit). The other solution is to set General = 0 and see as many orthopedic patients as possible. The final solution is to see as many of each type of patients as possible - this solution occurs at the intersection of the two constraints set at capacity. Let's illustrate. First, lets try a solution that includes all general patients, and no orthopedic patients. Looking at the two constraints, if Orth = 0, then we have enough surgery time to see 40 general patients, and enough therapy time to see 60 general patients. So, the most we can see is 40 general patients when Orth = 0 in order to keep a feasible solution. This solution has the following result: Z = 200 (40) + 300 (0) = $8,000 1) 2 (40) + 4 (0) = 80; slack = 0; constraint is binding 2) 2(40) + 12 (0) = 80 which is < 120; slack = 40; not binding Now let's try a solution that includes only orthopedic patients and no general patients. Looking at the two constraints, if General = 0, then we have enough surgery time to see 20 orthopedic patients, and enough therapy time to see 10 orthopedic patient. So, the most we can see is 10 orthopedic patient when General = 0 in order to keep a feasible solution. This solution has the following result: Z = 200 (0) + 300 (10) = $3,000 1) 2 (0) + 4 (10) = 40; which is < 80; slack = 40; not binding 2) 2 (0) + 12 (10) = 120; slack = 0; constraint is binding Now for the combination solution. First, set both constraints at their equality and solve the two equations for two unknowns. Do you remember how to do this? If not, don't worry about it since we will be using the software for problem solution. Here goes: 2 General + 4 Orth = 80 2 General + 12 Orth = 120 Subtract the second equation from the first: - 8 Orth = -40 Solve for Orth: Orth = 40/8 = 5 Substitute Orth = 5 in the first (or second) equation and solve for General: 2 General + 4 (5) = 80 2 General = 80- 4 (5) General = (80 - 20) / 2 General = 30 The profit for the solution Orth = 5, General = 30 is: Z = 200 (30) + 300 (5) = $7,500 Since the objective is to maximize net profit, the first solution with General = 40 and Orth = 0 is the optimal solution. Of course, some of the medical staff, such as the orthopedic surgeons may not be happy since they have no work with this surgery schedule. But that is not the fault of the method; it is a fault of the formulation. If the orthopedic surgeons want some work, they have to build that in to the constraint set, such as Orth > 10 as an added constraint. You won't be solving linear programming with such a "brute force" methodology as above, but with The Management Scientist. 6.2: Computer Solution and Interpretation We will work the clinic patient allocation model first. Here is the problem formulation: Maximize Z = 200 General + 300 Orth Subject to: 1) 2 General + 4 Orth < 80 2) 2 General + 12 Orth < 120 Using The Management Scientist Software Package To use The Management Scientist Inventory Module to solve for linear programming problems, click Windows Start/Programs/The Management Scientist/The Management Scientist Icon/Continue/Select Module 1 Linear Programming/OK/File/New and you are ready to load this example problem. In the next dialog screen, enter 2 for the Number of Decision Variables and 2 for the Number of Constraints. Then select Maximize for the Optimization Type; and then select OK. The next dialog screen first gives you the option to name the decision variables. X1 and X2 are the default names. I typed General over the X1 and Orth over the X2. Enter the objective function coefficients (200 and 300) right below the respective variable names. Finally, fill out the constraint matrix - it will look almost like the problem formulation above. Note that you are given just three choices for the relationship: , The < symbol is used for < constraints and the > symbol for > constraints. Here is the solution. Printout 6.1.1 LINEAR PROGRAMMING PROBLEM MAX 200 General+300 Orth S.T. 1) 2General+4Orth 5 Shift1 + Shift2 > 6 Shift2 + Shift3 > 10 Shift3 + Shift4 > 7 Shift4 + Shift5 > 4 Shift5 + Shift6 > 6 Can you come up with the optimal solution while I go off to run The Management Scientist solution? (five minute pause) .... I'm back. Did you figure that the department needs a minimum of 19 officers to meet the shift demand schedule, with 3 officers needed on Shift1, 3 on Shift2, 7 on Shift3, none on Shift4, 4 on Shift5 and 2 on Shift6? The computer solution in Printout 6.3.2 also shows negative 1's for the dual prices. Negative values for dual costs in minimization problem constraints means the objective function will "worsen" or get bigger when the right hand side of that constraint increases. For example, in constraint 2 if the right hand side of 10 were increased to 11, the objective function would worsen by 1, or we would need 20 officers instead of 19. Printout 6.3.2 LINEAR PROGRAMMING PROBLEM MIN 1Shift1+1Shift2+1Shift3+1Shift4+1Shift5+1Shift6 S.T. 1) 1Shift1+1Shift2>6 2) 1Shift2+1Shift3>10 3) 1Shift3+1Shift4>7 4) 1Shift4+1Shift5>4 5) 1Shift5+1Shift6>6 6) 1Shift1+1Shift6>5 OPTIMAL SOLUTION Objective Function Value = 19.000 Variable Value Reduced Costs -------------- --------------- ------------------ Shift1 3.000 0.000 Shift2 3.000 0.000 Shift3 7.000 0.000 Shift4 0.000 0.000 Shift5 4.000 0.000 Shift6 2.000 0.000 Constraint Slack/Surplus Dual Prices -------------- --------------- ------------------ 1 0.000 0.000 2 0.000 -1.000 3 0.000 0.000 4 0.000 -1.000 5 0.000 0.000 6 0.000 -1.000 OBJECTIVE COEFFICIENT RANGES Variable Lower Limit Current Value Upper Limit ------------ ------------ --------------- --------------- Shift1 0.000 1.000 1.000 Shift2 1.000 1.000 2.000 Shift3 1.000 1.000 1.000 Shift4 1.000 1.000 No Upper Limit Shift5 0.000 1.000 1.000 Shift6 1.000 1.000 2.000 RIGHT HAND SIDE RANGES Constraint Lower Limit Current Value Upper Limit ------------ --------------- --------------- --------------- 1 3.000 6.000 6.000 2 10.000 10.000 No Upper Limit 3 0.000 7.000 7.000 4 4.000 4.000 6.000 5 4.000 6.000 6.000 6 5.000 5.000 8.000 Multiperiod Production and Inventory Planning Model (Case 7) These models are used to determine a production, inventory and lost sales schedule so as to minimax cost or maximize profit over a multiperiod planning horizon. Here is some pertinent information for Allen Manufacturing Company (Problem 23, pp. 385-386 in the text). Table 6.3.2. Period Selling Price Per Unit Regular Production Cost Per Unit Overtime Production Cost per Unit Regular Production Capacity in Units Overtime Production Capacity in Units Ending Inventory Cost Per Unit Demand 1 $5.00 $2.80 $3.36 250 100 $0.50 500 2 $5.00 $2.90 $3.48 300 100 $0.50 300 3 $5.50 $3.00 $3.60 300 125 $0.55 400 Other pertinent planning information includes the lost sales cost of $4 per unit in any period. This amount accounts for the loss of customer good will, but does not include lost revenue which may also apply. The beginning inventory for period 1 is 100 units, and the firm would like to have at least 50 units in ending inventory for period 3 to prepare for period 4. The objective for this planning model is to determine the sales, regular time production amount, overtime production amount, ending inventory and lost sales for each of the production periods. We begin the formulation by identifying the many decision variables needed to answer the above objective. Each decision variable is measured in units (e.g., amount of sales in units). S1 = Sales (in units) in period 1 S2 = Sales (in units) in period 2 S3 = Sales (in units) in period 3 RP1 = Regular production (in units) in period 1 RP2 = Regular production (in units) in period 2 RP3 = Regular production (in units) in period 3 OP1 = Overtime production (in units) in period 1 OP2 = Overtime production (in units) in period 2 OP3 = Overtime production (in units) in period 3 BI = Beginning Inventory (in units) EI1 = Ending inventory (in units) in period 1 EI2 = Ending inventory (in units) in period 2 EI3 = Ending inventory (in units) in period 3 LS1 = Lost sales (in units) in period 1 LS2 = Lost sales (in units) in period 2 LS3 = Lost sales (in units) in period 3 We need a few more variables to complete model this production planning problem, although the variables are not decision variables and only appear in constraints. These include a variable for beginning inventory and three variables for demand: BI = Beginning Inventory (in units) D1 = Demand in period 1 (in units) D2 = Demand in period 2 (in units) D3 = Demand in period 3 (in units) The objective is to maximize profit, which is sales revenue minus costs (regular production, overtime production, inventory, and lost sales) for each of the three periods: Maximize Z = 5.00S1 - 2.80RP1 - 3.36OP1 - 0.50EI1 - 4.00LS1 + 5.00S2 - 2.90RP2 - 3.48OP2 - 0.50EI2 - 4.00LS2 + 5.50S3 - 3.00RP3 - 3.60OP3 - 0.55EI3 - 4.00LS3 For the constraints, let's start with meeting customer demand in each of the three period. Note that every production or service planning model will have customer demand as something that has to be met. The objective is still maximize profit but the demand has to be met in the long run or the firm doesn't survive. This may be a good time to note also that we need three separate constraints in this model, since D1+D2+D3 = 1200 may result in not meeting each period's demand. That is, if D1 = D2 = 0 and D3 = 1200 would be feasible to D1+D2+D3 =1200, but not to this problem. Also note that I made demand constraints as strict equalities. That is possible in this problem since inventory can handle excess demand, and lost sales shortages in demand. 1) Demand in period 1: D1 = 500 2) Demand in period 2: D2 = 300 3) Demand in period 3: D3 = 400 Every production or service planning model generally also have capacity constraints. In this case, we have both regular and overtime production capacity constraints. 4) Regular in period 1: RP1 < 250 5) Regular in period 2: RP2 < 300 6) Regular in period 3: RP3 < 300 7) Overtime in period 1: OP1 < 100 8) Overtime in period 2: OP2 < 100 9) Regular in period 3: OP3 < 125 Production planning problems need to include inventory constraints, unless inventory is not part of the picture (for example, companies that make perishable products may not have inventory constraints). Service companies obviously do not have inventory constraints. The beginning inventory is 100 units left over from last production cycle: that will be a strict equality. There is no requirement to have ending inventory from period 1 to 2, or from period 2 to 3. However, the firm stipulated that at the end of this three-period production cycle, it needs to have at least 50 units to go forward to the next cycle. The constraints are: 10) Beginning Inventory: BE = 100 11) Ending Inventory: EI3 > 50 Production planning models often include balance constraints that incorporate or model what happens when we produce more than we sell in periods 1, 2 and 3. When we produce more than we sell, we create ending inventory for the following period. Ending inventory results from starting with beginning inventory from the previous period, adding regular and overtime production, and subtracting sales. 12) Ending inventory in period 1: EI1 = BE + RP1 + OP1 - S1 13) Ending inventory in period 2: EI2 = EI1 + RP2 + OP2 - S2 14) Ending inventory in period 3: EI3 = EI2 + RP3 + OP3 - S3 We also need a set of constraints to model the situation that occurs when we produce less than we could have sold in periods 1, 2 and 3. When we produce less than we could have sold, we generate lost sales. 15) Lost Sales in period 1: LS1 = D1 - S1 16) Lost Sales in period 2: LS2 = D2 - S2 17) Lost Sales in period 3: LS3 = D3 - S3 Of course, all of the constraints have to be rewritten to put them in standard form for the linear program data input. That is shown in the computer input and output below. The computer output provides the solution values for the production planning decision and input variables, as well as rich sensitivity analysis information. Printout 6.3.3 LINEAR PROGRAMMING PROBLEM MAX 5S1+5S2+5.5S3-2.8RP1-2.9RP2-3RP3-3.36OP1-3.48OP2-3.6OP3-0.5EI1-0.5EI2-0.55EI 3-4LS1-4LS2-4LS3 S.T. 1) 1D1=500 2) 1D2=300 3) 1D3=400 4) 1RP1