Simulation and Optimization

Seth Berry

2021-08-26

Contents

1 Preface **5**

2 Introduction **7**
2.1 The Story 7

3 Linear Optimization **9**
3.1 Continuous Optimization 9

4 Integer Programming **19**

5 Simulation **21**

6 Nonlinear Optimization **23**

Chapter 1

Preface

You will find some form of this course in undergraduate and graduate business programs across the country. Instead of telling you what you should be looking for in such a class, it is easier to tell you what you shouldn't see:

1. Excel
2. Excel Add-ins
3. Palasaides

While the world runs on Excel, doing Simulation and Optimization in Excel will only ensure that you know Excel (can you use a VLookUp and sumproduct). Instead, the focus will be on understanding and implementing. You *will* be able to break problems down into the smallest parts and then roll those parts into objects. Throughout our time, we will get our hands dirty with some theory. These dives into theory will only serve to ease into greater understanding. In the end, however, the goal is application.

These techniques also serve as a nice introduction to programming in general, as they will allow us to scale from simple objects to very complex pipelines. Throughout our time here, we will mostly focus on using R. However, we don't live in a monolingual world anymore. To that end, we will see how some of these techniques translate to other languages (namely Python and Julia) and why we might want to consider one program over the other (speed, feature complete, ease, etc.).

Chapter 2

Introduction

In a world of exciting methods, simulation and optimization sit alone. Nobody touts how these things will change humanity. Nobody discusses how these methods can solve all of the world's problems. No…those conversations are reserved for things like statistical methods, machine learning, and the almighty AI! The secret, though, is that all of these fancy techniques do not exist without simulation and optimization.

Optimization is found in nearly every science: from nuclear medicine and biology, to electrical engineering and statistics, and beyond. While these fields are interesting on their own, our goal is to explore optimization in the context of business problem solving, with an occasional dive into how these techniques are used in techniques you learn in other courses.

Simulation is just as fundamental to the sciences as optimization. Nearly every event that happens is bound by some type of distribution and knowing that distribution allows us to test that event. What makes simulation so much fun is that you can program a version of the real world, run that program a few thousand times, and then generate a distribution of potential outcomes. This distribution will show us how common an event might be.

2.1 The Story

Meet Ali. Ali graduated from an Business Analytics program on the Atlantic coast in 2019 and made a smart career choice – accepting an offer to work as an analyst in the cannabis industry. The global cannabis industry has seen explosive growth during the last several years (topping 9 billion in 2020 and has a project compound annual growth rate of ~26% in America alone). While some of Ali's classmates (and family) questioned the decision, it was clear that it was an industry in need of some real analytics and Ali saw a real path towards making a difference for a business (after all, most people can't make a real difference in FAANGM). The Canadian cannabis industry has nearly a decade of maturity over the American cannabis industry, and American companies are looking to cover some of that lost ground. To that end, American companies are hiring people from all of the world to create the strongest possible teams. With diverse backgrounds and experiences, the general hope is that teams will function at the highest possible levels.

Ali belonged to a team with 3 other analysts: Alex, Jun, and Shashi. Ali was the youngest and least experienced of the entire group. What Ali lacked in experience, was more than made up by technical prowess. What Ali didn't know is that the analytics world has a dark secret. Throughout Ali's education, Python and R were touted as the most important languages in the world – they are, after all, where all of the exciting work happens. What Ali found, though, was that business analytics really runs on Excel and various add-ins.

Ali had a goal: to become the most valuable member of the team. Ali decided to take on anything the organization needed. It seemed like a good idea at first, but Ali found out that the Business Analytics program didn't really offer the proper preparation for what was to come.

Chapter 3

Linear Optimization

Ali's first task was to determine a marketing strategy. Both the Canadian and American cannabis industries are trying to normalize cannabis use (mainly through edibles and drinks) to women between the ages of 30 and 55. The working theory is that making cannabis use acceptable to this group will "allow" married men to also enjoy recreational cannabis use.

Ali's manager, Tolu, has asked to create a semi-automated system for determining advertisement spends. Thankfully, Tolu noted that Ali's coworker, Jun, has already been working in this space. Ali should be able to jump on Jun's work and make this system automated without much hassle.

3.1 Continuous Optimization

3.1.1 The Problem

What should have been an easy task became a nightmare. Ali didn't get a csv file with neatly defined columns and a clear outcome variable. No... Ali received this email (in which Jun was copied):

> Hi Ali,
>
> Here is what Rayan from Marketing needs:
>
> Instagram ads cost $50 dollars per hundred clicks
>
> TikTok ads cost $20 dollars per hundred clicks
>
> Over the last few weeks, we averaged about 1 female view for Instagram and 4 for TikTok. We need at least 80 female views in total for the coming week.
>
> We don't really do as well with men; we saw just about 1 average male view for both Instagram (.9) and TikTok (.8). We are really hoping to get at least 40 for the coming week.
>
> Where should we buy ads for the coming week?
>
> All my best,
>
> Tolu

3.1.2 From Words To Formulas

In typical analyst fashion, Ali responded with, "No problem!", and started digging through old course notes. Unfortunately, nothing looked like this problem. Ali decided that a cup of coffee with Jun was the way to go. Before a coffee invite even went out, Jun sent Ali a copy of the legacy Excel sheet... completely full of Excel equations and Solver boxes.

Ali had no idea what was going on in the sheet – there were *sumproduct* formulas, *vlookups*, and other strange things. Ali asked Jun to go over the sheet and the problem together, and Jun most graciously agreed.

Jun helped Ali break the problem down into small pieces. “The first question”, Jun said, “is what are the variables and what are their values?”

Ali thought for a minute and decided that there are two variables to this problem: Instagram and TikTok. “Correct!”, said Jun, “and what values do Instagram and TikTok have?” Ali went back to the email and saw:

> Instagram ads cost $50 dollars per hundred clicks
>
> TikTok ads cost $20 dollars per hundred clicks

“Awesome! The value for Instagram is 50 and the value for TikTok is 20.”, Ali said. “And what specifically are those?”, Jun asked. Ali wasn’t sure, but said, “ad costs”. “Now, let me show you something.”, and Jun wrote this on a piece of paper:

$$\text{ad cost} = 50_{instagram} + 20_{tiktok}$$

“What else do we need?”, asked Jun. Ali thought for a minute and said, “We need to get the rest of the information into the problem!”

> Over the last few weeks, we averaged about 1 female view for Instagram and 4 for TikTok.
>
> We don’t really do as well with men; we saw just about 1 average male view for both Instagram (.9) and TikTok (.8)

“Let’s put that into our problem”, and Jun was back to writing:

$$\begin{aligned} \text{ad cost} = 50_{instagram} + 20_{tiktok} \\ 1_{instagram} + 4_{tiktok} \\ .9_{instagram} + .8_{tiktok} \end{aligned}$$

“Did Rayan have an specific needs for those men and women?”, asked Jun. Again, Ali looked at the email and saw:

> We need at least 80 female views in total for the coming week.
>
> We are really hoping to get at least 40 for the coming week.

“So,”, Ali began, “We need at least 80 views for women and 40 views for men. We could have more for both, though. . . it is just the baseline.”

“Excellent! Check this out”, and Jun added the following:

$$\begin{aligned} \text{ad cost} = 50_{x1} + 20_{x2} \\ \text{women} = 1_{instagram} + 4_{tiktok} \geq 80 \\ \text{men} = .9_{instagram} + .8_{tiktok} \geq 40 \end{aligned}$$

“And we are almost there!”, Jun smiled and then asked, “What would we want to do with cost: spend as much as possible or as little as possible?” “Oh”, Ali said, “that’s easy: we definitely want to minimize our cost.”

$$\begin{aligned} \text{Minimize:} \\ \text{ad cost} = 50_{instagram} + 20_{tiktok} \\ \text{Subject to:} \\ \text{women} = 1_{instagram} + 4_{tiktok} \geq 80 \\ \text{men} = .9_{instagram} + .8_{tiktok} \geq 40 \end{aligned}$$

"Here's the last question", Jun said, "Could we buy a negative number of ads?". "Absolutely not", Ali said.

"We have this now", and Jun showed Ali his paper

$$\begin{aligned}\text{Minimize:}\\ \text{ad cost} = 50_{instagram} + 20_{tiktok}\\ \text{Subject to:}\\ \text{women} = 1_{instagram} + 4_{tiktok} \geq 80\\ \text{men} = .9_{instagram} + .8_{tiktok} \geq 40\\ \text{instagram, tiktok} \geq 0\end{aligned}$$

"Now that we have this put completely together, we need to break it down", Jun laughed.

"We know that this is a **minimization** problem", Jun said and continued, "and we know that we have two **variables**: Instagram and TikTok. You may hear people call these **objective values**."

Jun carried on, "We also know that we have some rules to follow for our problem. These rules are called **constraints**. Think of these constraints as rules that reflect reality".

"Got it!", exclaimed Ali.

"Let's see them again", Jun said:

$$\begin{aligned}\text{Subject to:}\\ \text{women} = 1_{instagram} + 4_{tiktok} \geq 80\\ \text{men} = .9_{instagram} + .8_{tiktok} \geq 40\end{aligned}$$

"This whole thing is the **constraint matrix** and we can break it down into its component parts!", beamed Jun.

"Let's start with the **left-hand side** of the constraint matrix, which you might hear referred to as the **A matrix**.

$$\begin{aligned}1_{instagram} + 4_{tiktok}\\ .9_{instagram} + .8_{tiktok}\end{aligned}$$

"We have 4 values, spread across 2 columns and 2 rows.", Jun said, "Just like a normal table". Jun continued, "Next we come to the **directions**…those things look like inequalities, but we will also probably encounter some equalities too".

"Here, we just have a simple **vector** of those signs.", Jun wrote:

$$\begin{gathered}\geq\\ \geq\end{gathered}$$

"Last thing, I promise.", Jun said: "The **right hand side**, those values that we need to achieve, are referred to as the **marginal values**."

$$\begin{gathered}80\\ 40\end{gathered}$$

"Tolu said that you were going to program these in Q or Boa, or something like that. I can't help you there, but let me know if I can do anything else for you", Jun said and walked back to the office.

Ali felt better, but getting all of that information into R was going to be a little bit tricky.

3.1.3 Application

Ali was feeling pretty good after all of this! As soon as the computer was unlocked, StackOverlow came to the rescue – a user called Not_Prof_Berry had answered a few questions about linear programming with R.

It seemed like Ali was going to need a package called `linprog`:

```
# install.packages('linprog')

library(linprog)
```

The specific function is `solveLP`, but Ali saw that it needed some objects to be created first: `cvec`, `bvec`, `Amat`, and `const.dir`. Ali remembered a common mantra among professors – "Read the flipping manual!". After reading the helpfile, Ali determined that the cvec object needed to contain the objective values:

```
objective_values <- c(50, 20)
```

Ali then figured out that bvec came from the right-hand side of the constraint matrix (the values out in the margin of the constraint matrix):

```
constraint_values <- c(80, 40)
```

The Amatrix felt a little bit tricky. It definitely needed to be the constraint matrix, but it was somewhat tough to get into the right shape. Ali tried a few things:

```
constraint_matrix <- rbind(c(1, 4),
                           c(.9, .8))

constraint_matrix2 <- matrix(c(1, 4, .9, .8),
                             ncol = 2, nrow = 2,
                             byrow = TRUE)
```

Both returned a matrix:

```
str(constraint_matrix)
```

```
 num [1:2, 1:2] 1 0.9 4 0.8
```

```
str(constraint_matrix2)
```

```
 num [1:2, 1:2] 1 0.9 4 0.8
```

Which Ali knew was needed for function to work properly.

Finally, Ali made a character vector of const.dir (i.e., the constraint directions):

```
constraint_directions <- c(">=", ">=")
```

With those 4 objects, Ali was ready to solve the problem!

```
solved_model <- solveLP(cvec = objective_values,
                        bvec = constraint_values,
                        Amat = constraint_matrix,
                        maximum = FALSE,
                        const.dir = constraint_directions)
```

With the model solved, Ali needed to grab some information: the recommended values for Instagram and TikTok, and how much money it was going to cost.

First, how much money was this going to cost:

```
solved_model$opt
```

```
[1] 1000
```

Got it. $1000 is the total spend.

Second, what is the marketing mix:

```
solved_model$solution
```

```
 1  2
 0 50
```

That gives 0 ads for Instagram and 50 ads for TikTok! There is no way that is correct. Everyone knows that a 0 for an answer means that there has to be a problem. How could Ali take this solution to Tolu? Ali figured the best course of action would be to check the solution with Jun.

3.1.4 Theory

Like all early-career analysts, Ali was feeling beaten – going back to Jun so quickly felt like a failure. Like all experienced analysts, Jun was only too happy to help and explain what was happening.

Jun reminded Ali of the complete notation they had created together.

$$\begin{aligned}\text{Minimize:}\\ \text{ad cost} = 50_{instagram} + 20_{tiktok}\\ \text{Subject to:}\\ \text{women} = 1_{instagram} + 4_{tiktok} \geq 80\\ \text{men} = .9_{instagram} + .8_{tiktok} \geq 40\\ \text{instagram}, \text{tiktok} \geq 0\end{aligned}$$

“Let’s break this down a little bit”, and turning to the whiteboard, Jun wrote:

$$1_{instagram} + 4_{tiktok} = 80$$

Can be solved with:

$$(instagram = 0, tiktok = 20) \text{ or } (instagram = 80, tiktok = 0)$$

And:

$$.9_{instagram} + .8_{tiktok} = 40$$

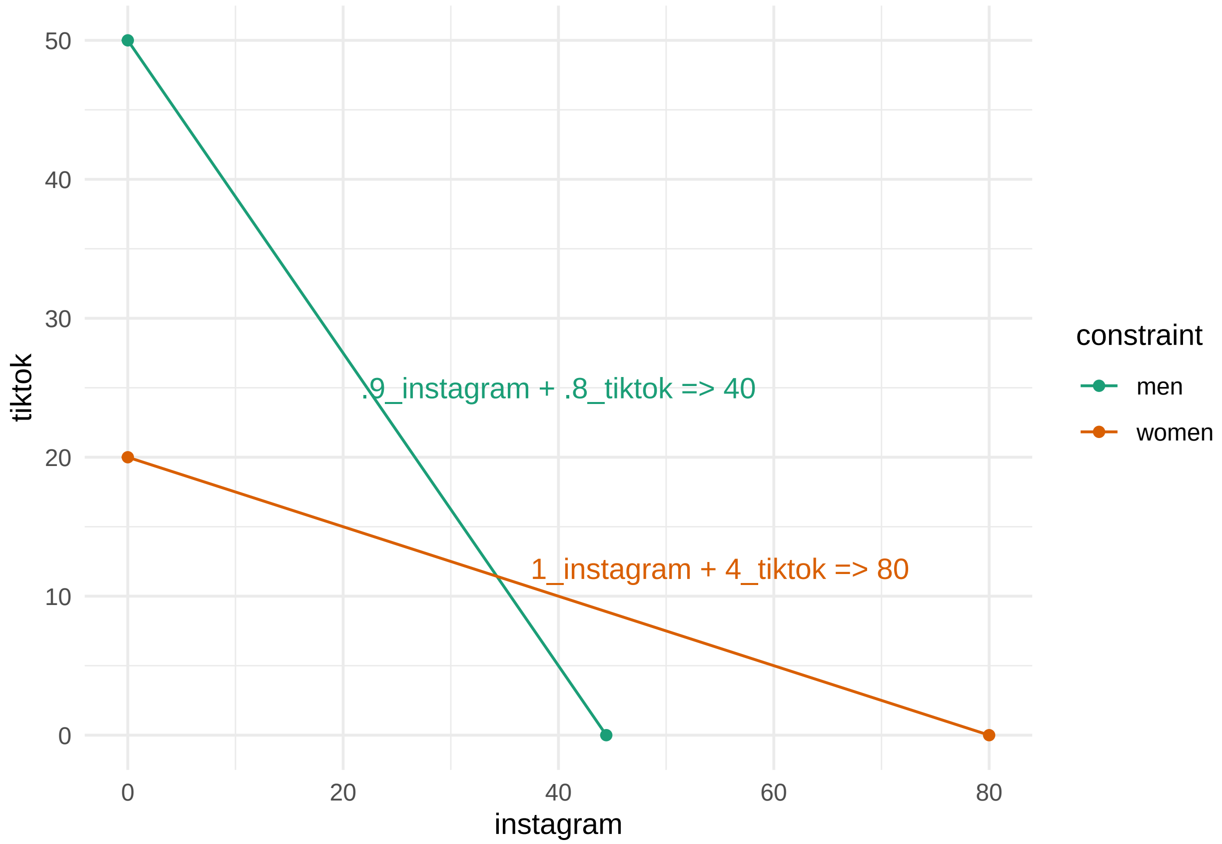
Can be solved with:

$$(instagram = 44.44, tiktok = 0) \text{ or } (instagram = 0, tiktok = 50)$$

“It’s okay if you don’t remember or didn’t take linear algebra”, Jun noted, “just know that we are solving these equations to obtain a set of points.”

“Okay”, Ali nodded, “but what do we do with those points?”

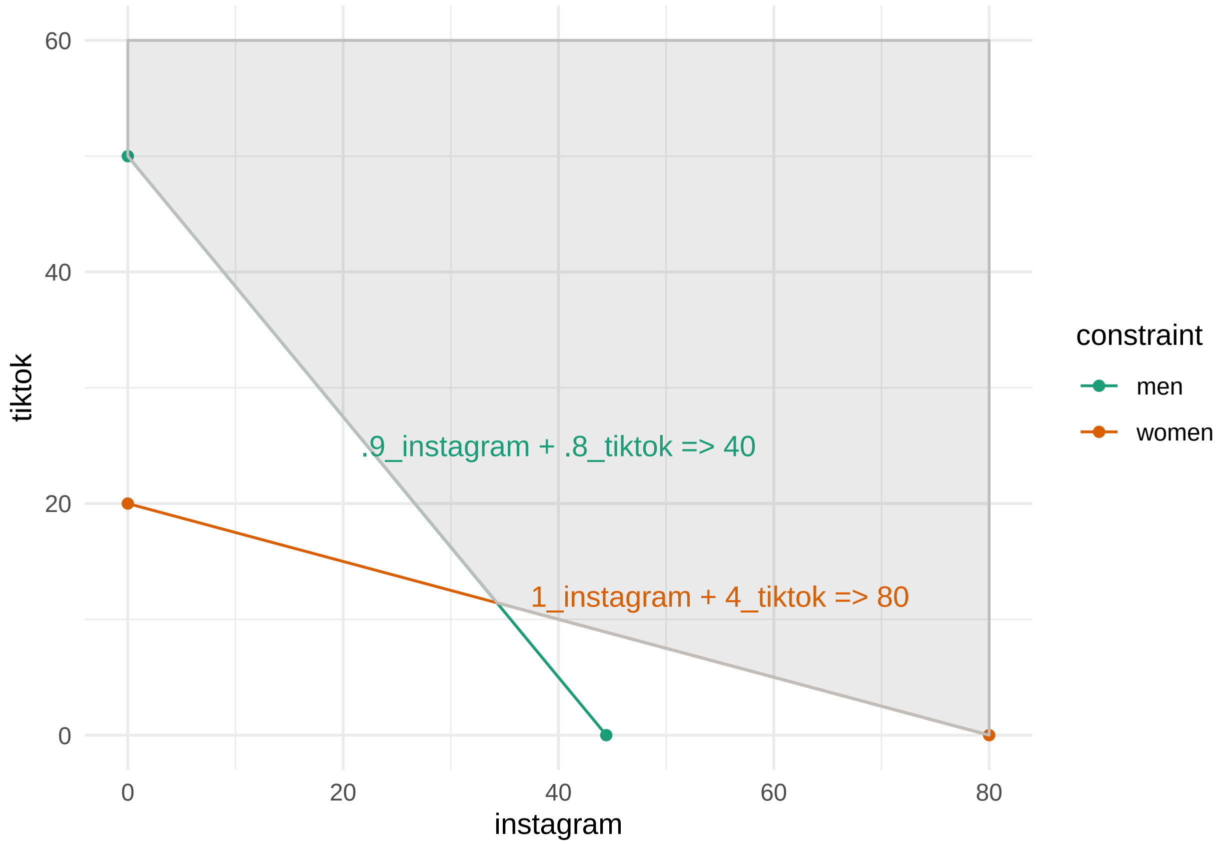
“Plot them”, and Jun went back to writing:



"Now that we have those lines plotted, we can clearly see where we might find our answer!", beamed Jun.

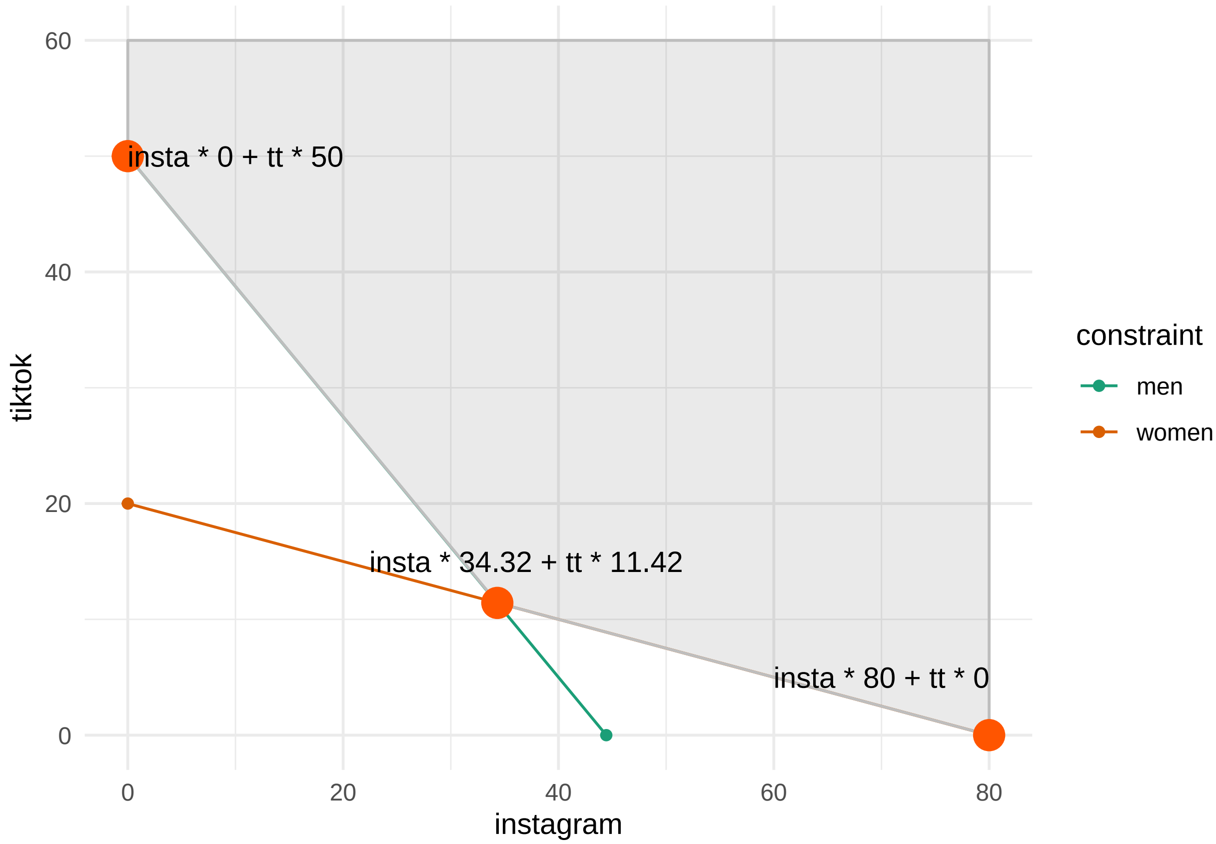
"Umm…it's a little fuzzy", admitted a puzzled Ali.

"Completely to be expected", Jun smiled. "Let's find the **feasible region** – this is the place where our answer lives!"



"We could go through and try every single set of points in this shaded area, but we would never finish and we would be wasting our time", Jun chuckled and continued, "We already know that we are looking for the values that minimize our solution, so we can completely ignore every point that doesn't sit on our lines."

Jun made a few circles on the plot and said, "These points will give us our answer!"



"This is called the **extreme point theorem** and it basically says that our **optimal** solution has to rest somewhere in the extreme points of the feasible region", said Jun, "and it makes life so much easier for finding our answer."

"We can just do the simple math now", and Jun wrote"

```
insta_cost = 50

tt_cost = 20

insta_cost * 0 + tt_cost * 50
```

```
[1] 1000
```

```
insta_cost * 34.32 + tt_cost * 11.42
```

```
[1] 1944.4
```

```
insta_cost * 80 + tt_cost * 0
```

```
[1] 4000
```

"Which of those is the smallest value?", Jun asked.

The conference room glowed with Ali's excitement! "I got it now!", Ali exclaimed. "We can get our optimal value of 1000 by purchasing 50 ads on TikTok and 0 on Instagram! The solution was correct!"

"Ahhh!", Jun started laughing, "it is definitely the optimal solution, but do you *really* think that it's the correct solution?" Jun shook his head and continued laughing, "How do you think Tolu is going to take the advice to not put anything at all on Instagram?"

via GIPHY

Ali's mind was sufficiently wrecked. How could an optimal answer not be the correct answer? Stupid analytics – nothing can ever be easy.

"What's the best path forward, then?", Ali asked Jun. "Simple", Jun replied, "ask how much they want to put on Instagram and that becomes a constraint!"

This was to be an important lesson for Ali: analytics tasks are never a one-shot deal. Clarity needs to be sought before most work can actually happen.

After a quick email exchange with Rayan from Marketing, Ali found out that at least 10 ads were needed for Instagram.

3.1.5 ClassOverflow

Let's spend some time helping Ali. We need to do two things: 1) specify an appropriate model and 2) solve it.

We will do this two different ways; both are good to know, but I'd imagine that you will find one to be more valuable than the other.

3.1.6 Using Python

A great chunk of Ali's coursework was in R, with just some excursions into Python. For statistics, R reigns supreme (but statsmodels in Python is really pretty solid). For machine learning, take your pick (only fanboys speak in absolutes about one being better than the other – both have their pros and cons). Linear programming is a bit different. R has some clear advantages in terms of flexibility, but Google has put effort towards implementing their GLOP solver in Python (among other languages).

The `pulp` package is going to look different than what we saw in R, but there are some definite improvements in expressing our model:

```python
from pulp import *

model = LpProblem(name = "test-model",
  sense = LpMinimize)

x = LpVariable(name = "instagram", lowBound = 0)
y = LpVariable(name = "tiktok", lowBound = 0)

model += (1 * x + 4 * y >= 80, "women")
model += (.9 * x + .8 * y >= 40, "men")

obj_func = 50 * x + 20 * y
model += obj_func

model

status = model.solve()

model.objective.value()

x.value()
y.value()

for var in model.variables():
  print(f"{var.name}: {var.value()}")
```

We really aren't breaking our problem down into small objects here. Instead, all we really need to do is to take our math form and pop that into our `model` – pretty easy stuff.

Finally, here is Google's OR tools. It is the current SoTA for optimization (how do you think Google gets people navigated). You'll notice that we aren't really doing anything too different than what we saw with `pulp`:

```
from ortools.linear_solver import pywraplp

solver = pywraplp.Solver.CreateSolver('GLOP')
x = solver.NumVar(0, solver.infinity(), 'instagram')
y = solver.NumVar(0, solver.infinity(), 'tiktok')

solver.NumVariables()
```

```
2
```

```
solver.Add(1 * x + 4 * y >= 80)
```

```
<ortools.linear_solver.pywraplp.Constraint; proxy of <Swig Object of type 'operations_research::MPConst
```

```
solver.Add(.9 * x + .8 * y >= 40)
```

```
<ortools.linear_solver.pywraplp.Constraint; proxy of <Swig Object of type 'operations_research::MPConst
```

```
solver.NumConstraints()
```

```
2
```

```
solver.Minimize(50 * x + 20 * y)

status = solver.Solve()

solver.Objective().Value()
```

```
999.9999999999999
```

```
x.solution_value()
```

```
0.0
```

```
y.solution_value()
```

```
49.99999999999999
```

Chapter 4

Integer Programming

You have gone just a little too far.

You will find the previous chapter to be more enlightening at this time.

Chapter 5

Simulation

Why are you looking this far ahead?

Focus on the present instead.

Chapter 6

Nonlinear Optimization

While you have clearly taken a linear line to get to this point, you should probably go backwards across your linear line.