IE306.01
ASSIGNMENT 1

Ahmet Fırat GAMSIZ	2020400180
Ceyhun SONYÜREK	2020400258
Işıl Su KARAKUZU	2020400144
Yusuf AYGÜN	2020400033
Furkan ŞENKAL	2020400249

1. Introduction **2**
2. Structure **3**
3. Results **5**
4. Opinions and Comments **5**

1. Introduction

Let's start with the given formulations. Our manager estimated demand to be normally distributed, with mean 12000 and std 2750. Also, we have the fixed costs given in the documentation. Using these costs we can easily understand the profit function. If we produce less than demand, then we will sell all the products. So, our profit will be selling price times production quantities (since we are able to sell all produced goods) minus variable production cost times production quantity minus the fixed cost that always exists.

However if we produce more than demand then we won't be able to sell the whole production. Then we need to multiply the selling price with demand and decrement the variable production cost times production quantity and also, add the salvage value times the quantity of units we weren't able to sell (so we will be getting some money back from those extra products). And finally decrement the fixed cost again.

We are looking for the production quantity that will maximize this profit function. For this we used discrete values ranging over 5000 to 16000 including increasing by 1000. Our algorithm includes generating random demands using the given normal distribution for 1000 trials. We compute the profit for each demand and estimate the expected profit. Then we repeat from the generation for the next production quantity.

2. Structure

To implement the algorithm explained in the introduction, we aimed to harness the power of automation. Thus, we created the following Python script:

```
import openpyxl

my_excel = openpyxl.load_workbook('IEHW1.xlsx')
my_sheet = my_excel.active

for n in range(1000):
   my_sheet.cell(n+1,1,5000)
   my_sheet.cell(n+1,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+1,3,
f'=IF(B{n+1}>=A{n+1},125*A{n+1}-100*A{n+1}-100000,125*B{n+1}-100*A{n+1}+27.5*(A{n+1}-B{n+
1})-100000)')
   my_sheet.cell(n+1001,1,6000)
   my_sheet.cell(n+1001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+1001,3,
f'=IF(B{n+1001}>=A{n+1001},125*A{n+1001}-100*A{n+1001}-100000,125*B{n+1001}-100*A{n+1001}
+27.5*(A{n+1001}-B{n+1001})-100000)')
   my_sheet.cell(n+2001,1,7000)
   my_sheet.cell(n+2001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+2001,3,
f'=IF(B{n+2001}>=A{n+2001},125*A{n+2001}-100*A{n+2001}-100000,125*B{n+2001}-100*A{n+2001}
+27.5*(A{n+2001}-B{n+2001})-100000)')
   my_sheet.cell(n+3001,1,8000)
   my_sheet.cell(n+3001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+3001,3,
f'=IF(B{n+3001}>=A{n+3001},125*A{n+3001}-100*A{n+3001}-100000,125*B{n+3001}-100*A{n+3001}
+27.5*(A{n+3001}-B{n+3001})-100000)')
   my_sheet.cell(n+4001,1,9000)
   my_sheet.cell(n+4001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+4001,3,
f'=IF(B{n+4001}>=A{n+4001},125*A{n+4001}-100*A{n+4001}-100000,125*B{n+4001}-100*A{n+4001}
+27.5*(A{n+4001}-B{n+4001})-100000)')
   my_sheet.cell(n+5001,1,10000)
   my_sheet.cell(n+5001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+5001,3,
f'=IF(B{n+5001}>=A{n+5001},125*A{n+5001}-100*A{n+5001}-100000,125*B{n+5001}-100*A{n+5001}
+27.5*(A{n+5001}-B{n+5001})-100000)')
   my_sheet.cell(n+6001,1,11000)
   my_sheet.cell(n+6001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+6001,3,
```

```
f'=IF(B{n+6001}>=A{n+6001},125*A{n+6001}-100*A{n+6001}-100000,125*B{n+6001}-100*A{n+6001}
+27.5*(A{n+6001}-B{n+6001})-100000)')
   my_sheet.cell(n+7001,1,12000)
   my_sheet.cell(n+7001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+7001,3,
f'=IF(B{n+7001}>=A{n+7001},125*A{n+7001}-100*A{n+7001}-100000,125*B{n+7001}-100*A{n+7001}
+27.5*(A{n+7001}-B{n+7001})-100000)')
   my_sheet.cell(n+8001,1,13000)
   my_sheet.cell(n+8001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+8001,3,
f'=IF(B{n+8001}>=A{n+8001},125*A{n+8001}-100*A{n+8001}-100000,125*B{n+8001}-100*A{n+8001}
+27.5*(A{n+8001}-B{n+8001})-100000)')
   my_sheet.cell(n+9001,1,14000)
   my_sheet.cell(n+9001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+9001,3,
f'=IF(B{n+9001}>=A{n+9001},125*A{n+9001}-100*A{n+9001}-100000,125*B{n+9001}-100*A{n+9001}
+27.5*(A{n+9001}-B{n+9001})-100000)')
   my_sheet.cell(n+10001,1,15000)
   my_sheet.cell(n+10001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+10001,3,
f'=IF(B{n+10001}>=A{n+10001},125*A{n+10001}-100*A{n+10001}-100000,125*B{n+10001}-100*A{n+
10001}+27.5*(A{n+10001}-B{n+10001})-100000)')
   my_sheet.cell(n+11001,1,16000)
   my_sheet.cell(n+11001,2, '= 12000 + 2750 * NORMSINV(RAND())')
   my_sheet.cell(n+11001,3,
f'=IF(B{n+11001}>=A{n+11001},125*A{n+11001}-100*A{n+11001}-100000,125*B{n+11001}-100*A{n+
11001}+27.5*(A{n+11001}-B{n+11001})-100000)')



my_excel.save('IEHW1.xlsx')
```

The script creates the excel file, then for each production quantity it runs a single trial. Trials are done by first entering production quantity to the cell and then using the formula =12000 + 2750 * NORMSINV(RAND()). This formula inserts our guessed demand to the following cell. Then we use =IF(B{n+1}>=A{n+1},125*A{n+1}-100*A{n+1}-100000,125*B{n+1}-100*A{n+1}+27.5*(A{n+1}-B{n+1})-100000)') formula to estimate the profit and insert it to the next cell. Note that we change n's to read from and write to appropriate cells. Then we execute this script and generate our values. After that we open our excel file to calculate the average values. We use =SUM(C1:C1000)/1000 formula to calculate and insert the average value for the trials with 5000 unit products. Then we adjust the cell range accordingly and calculate other average values too.

3. Results

5000 AVG	6000 AVG	7000 AVG	8000 AVG	9000 AVG	10000 AVG	11000 AVG	12000 AVG	13000 AVG	14000 AVG	15000 AVG	16000 AVG
24835,61	47965,14	71557,86	91838,64	106911,7	110649,8	106241	92866,58	66102,41	22029,11	-38736,3	-101579

As one can see, the highest expected profit is gained on producing 10000 units, and the production quantities seem to follow a bell shaped distribution. Trying to fine-tune things between making 9000 and 11000 units might not make an important increase in profit. While this analysis can be easily done with software tools, a better optimization would come from a better demand estimation. The demand varies a lot due to a big standard deviation.

4. Opinions and Comments

Instead of relying on a simple distribution, one can consult industry-standard demand prediction methods such as Prophet. It could provide more accurate demand forecasts, and we could also consider fluctuations due to seasons and holidays, etc. Following these trends would enable us to easily estimate the maximum production quantity before the next production season and produce accordingly.