Group 6: Insight of Casterbridge Case Study

Aneesh Agarwal[1], Leo Kurata[2], Nayla Rosihan[3], Vu Duc Dang[4]

[1] 34654313, aaga0022@student.monash.edu
[2] 34576452, lkur0008@student.monash.edu
[3] 33173273, nros0021@student.monash.edu
[4] 32698178, vdan0010@student.monash.edu

Abstract :

We are a highly proficient team in business modelling on business thanks to ETF3480 - ETF5480 unit. Recently, we had a client, Casterbridge Bank, who is interested in how many new analysts the bank should make offers to in the London office. Our mission is to analyse which strategy is the best for the client to adopt. This paper contains findings from our analysis with simulation to compare four scenarios. We conclude that the 'hiring every six months and at each time allowing new graduates to choose to join the company two months later' strategy (called Scenario 4 in this paper) is the best for annual earnings. After pointing out that the simulation data we got can justify this result from multiple aspects, we also discuss business recommendations and other considerations in the second half.

* This report has Errata.

1 Introduction

Casterbridge Bank, a recently established investment bank in London, has quickly developed a reputation for outstanding client service and attracting top-tier clients due to its dynamic approach. As the bank shifts its focus towards stabilizing and optimizing internal management to support sustained growth and efficiency, this report delivers an in-depth analysis of various staffing strategies over a 12-year period. It aims to identify the most cost-effective approaches that align with the bank's transition from rapid expansion to more measured growth. Utilizing detailed simulation models, the analysis evaluates different staffing scenarios and their impacts on the bank's financial health and operational efficiency. The report encompasses several critical areas: it outlines how the simulation was set up, detailing the methodologies and statistical methods used; presents the results from the simulations, highlighting which scenarios were most cost-effective; discusses these results and offers strategic insights and recommendations for adjusting staffing policies; and concludes with a summary of key findings and suggestions for further research and potential strategy refinements. Through these insights, the report equips Casterbridge Bank's management with actionable recommendations that facilitate informed decision-making, helping the bank refine its staffing approaches to mitigate financial risks and enhance operational efficiency in a competitive financial environment.

1.1 Background and Problem Definition

Current Strategy

Historically, Casterbridge Bank's strategy for hiring analysts has been straightforward and somewhat reactive. Based on feedback from division managers, there has been a recurring issue with the alignment of analyst hiring targets—sometimes too low and at other times resulting in surplus analysts without adequate projects. This misalignment leads to significant financial implications, either through the increased costs of idle labor or the inefficiencies of deploying transferred staff from other locations, who are typically 60% less efficient due to various factors including language, cultural differences, and retraining needs.

Staffing Fluctuations

The demand for analysts at Casterbridge is highly variable, influenced by both market conditions and internal project flows. This fluctuation poses a substantial challenge as it requires a flexible yet precise approach to staffing. The bank faces periods of both overstaffing, where analysts remain unutilized, and understaffing, where there is a scramble to meet client needs, often by pulling in less efficient resources from other offices.

Problem Statement

The primary challenge Casterbridge Bank faces is optimizing its staffing strategy to better align with the fluctuating demand for analyst services. The aim is to minimize financial losses due to overstaffing and underutilization and to avoid the high costs associated with under-provisioning and the resultant need for inefficient staffing solutions. This report will explore various strategic adjustments to determine an optimal balance that maximizes profitability while maintaining the bank's high standards of client service.

1.2 Data Sources

The analysis is based on historical staffing and workload data compiled by the bank's HR department over five years. This data includes monthly records of analyst staffing levels and workload variations, enriched with economic indicators affecting market conditions.

Limitations

- Confidentiality: Constraints prevent the sharing of detailed individual performance metrics and specific client engagement data, limiting the depth of personalized analysis.
- Historical Bias: Data reflect past conditions under different strategic directives, potentially limiting applicability to current and future conditions.
- Historical Bias: Data reflect past conditions under different strategic directives, potentially limiting applicability to current and future conditions.
- Data Completeness: Gaps exist in the data, especially regarding transient staff and short-term contractors.

Assumptions

- Demand Patterns: Analyst demand is assumed to peak in March and September, aligning with key market activities and client financial cycles.
- Cost Structures: Direct (salaries and bonuses) and indirect costs (training, health insurance) are assumed constant.
- Staffing Efficiency: Transferred or pooled analysts are assumed to operate at 60% efficiency relative to local staff.
- Economic Impact: Economic conditions are modelled as a normal distribution, influencing the variability in demand for analyst services.

1.3 Paper Organisation

This report delivers an in-depth analysis of various staffing strategies over a 12-year period. It aims to identify the most cost-effective approaches that align with the bank's transition from rapid expansion to more measured growth. Utilizing detailed simulation models, the analysis evaluates different staffing scenarios and their impacts on the bank's financial health and operational efficiency. The report encompasses several critical areas: it outlines how the simulation was set up, detailing the methodologies and statistical methods used; presents the results from the simulations, highlighting which scenarios were most cost-effective; discusses these results and offers strategic insights and recommendations for adjusting staffing policies; and concludes with a summary of key findings and suggestions for further research and potential strategy refinements. Through these insights, the report equips Casterbridge Bank's management with actionable recommendations that facilitate informed decision-making, helping the bank refine its staffing approaches to mitigate financial risks and enhance operational efficiency in a competitive financial environment.

2 Methods

2.1 Comprehension about setting description

2.1.1 Setting Our client is Susan, an HR director at the Casterbridge Bank. She began the annual recruiting drive for analysts at the bank's London office. The problem is that the HR could not manage the number of analysts in that office appropriately last year, that is, it is often understaffed and overstaffed. Therefore, the client is trying to keep the analysts in the level of demand by recruiting the "right" number of analysts from the main pool of graduating students. The bank also wants to take into account many internal and exogenous factors such as analyst demands depending on a seasonal cycle, hiring (e.g., the retention rate, acceptance rate of new graduates), and economic shock (e.g., economic growth, market volatility affecting on demand).

The company charges its customers fixed fees or percentages of the value of deals or issues, based on an internal billing rate of $10,000 per month for an analyst's time. Meanwhile, analysts work any hours necessary to serve their clients' interests for an monthly salary of around $4,000. Additional analyst's labour cost to company is typically $24000 per month in indirect support costs such as training, health insurance, and other employee benefits.

When the demand for analyst time on client projects exceeded the number of analysts in the London office, the operations department would temporarily transfer staff from other offices in Europe and/or pool analysts from different departments within the London office to make up for the shortage. Nonetheless, it is already well known that transferred employees tended to be less productive than those working in their home office usually by a factor of 60%.

Under these situations, the client currently makes offers after the April recruiting round and newly hired analysts were asked to start work on July 1. Now, it is thinking of a creative hiring strategy which might be much more effective than a current one. Here are two ideas the bank provided.

- What if it allows new recruits to choose to start at the beginning or at the end of the summer (either July 1 or September 1). Then any remaining imbalance might be less costly. We expect about half of the new recruits chose to start work in July. Of the remaining half who chose to delay their start-date to September, between 70% and 100% fulfilled their promise to join the firm.
- The bank makes an offer in December in order to capture students who graduate at mid-year. Mid-year recruiting, if done correctly, could help to alleviate analyst staffing shortages. In fact, the option to do December recruiting might even allow Casterbridge to hire fewer analysts for July 1 and then make a mid-year correction if need be. Analysts hired in the December round would be asked to start work at the beginning of January.

2.1.2 What we need to solve We can observe two essences to be discussed from the ideas in the previous section: (i) whether the company should recruit new graduates also in January, and (ii) whether the option of when new graduates start their careers improves the efficiency or not. We, thus, consider four scenarios in total, each comprising a combination of two elements. On the top of that, the follows are questions we are supposed to solve and report the answer for.

1. Did it make sense to offer recruits the option of starting at the beginning of September?
2. Should the bank be more radical and hire twice each year instead of only once?
3. Which strategy was best for the bank? How could we show our chairman the monetary implications of each strategy?

It goes without saying that what motivates them is that both understaffing and overstaffing have a financial impact on them. Therefore, it is rational to compare annual earnings and choose a strategy entailing the highest performance among the given multiple scenarios.

2.2 Construction the Simulation

2.2.1 Notation $i \in I$ is the non-negative integer and denotes a certain period mainly used as an index. We suppose $i = 0$ indicates April and foresee T months ahead. Define the bank analysts' demand and supply to consider a balance between them. We estimate the demand in month i, denoted by D_i, relying on the historic average analyst demand H_i in that month i as a baseline. We have already obtained the list of H_i.

We assume that there are multiple shocks. As analyst demand levels would come from unexpected economic growth or decline over the whole year, define a random variable $X \in [-1, 1]$ as a

Table 1. The historical data of the number of analysts

Month	Analyst Demand
June	75
July	70
August	70
September	110
October	105
November	90
December	65
January	80
February	90
March	120
April	105
May	95
TOTAL	1075
Average per month	90

percentage of unanticipated economic growth per year[1], which follows the normal distribution $\mathcal{N}(0, 0.05)$[2]. Also, to consider the fluctuation in both directions, a random variable $Y_i \in [-1, 1]$ to follow the normal distribution $\mathcal{N}(0, 0.10)$ denotes a random noise in demand in month i compared to expected historic level in that month. In the end, we calculate the value $D_i \in \{D_i\}_{i=1}^T \equiv D$ as follows.

$$D_i = H_i(1 + X)(1 + Y_i) \tag{1}$$

On the other hand, supply side of analysts is complicated to calculate since we have to think about both currently employees and new hired graduates. Firstly, some of the currently employees are supposed to quit the company. Since the bank observes some remarkable seasonal cycles, we can consider the retention rate R_i follows some given uniform distribution.

Table 2. Monthly Average Retention Rate

Months	Prob. Distribution
January and September	$R_i \overset{\text{i.i.d.}}{\sim} \mathcal{U}(0.80, 1)$
May, June, July and August	$R_i \overset{\text{i.i.d.}}{\sim} \mathcal{U}(0.95, 1)$
February, March, April, October, November and December	$R_i \overset{\text{i.i.d.}}{\sim} \mathcal{U}(0.90, 1)$
Simple average	0.95

Next, when it makes offers, the bank takes into account both the number of analysts it needs in the month and the acceptance rate of graduates who get an offer. The number of job offers can be obtained based on the equation (2).[3]

$$\begin{aligned} &\text{Average demand} \\ &=(\text{estimated \# of analysts as of 1st July} + \text{\# of accepted offers}) \\ &\quad \times (\text{average annual retention rate}). \end{aligned} \tag{2}$$

[1] Although X is an annual value, we can here assume it as a monthly economic growth rate.

[2] The second element inside the parenthesis is the standard deviation, rather than variance in this report.

[3] E.g., $90 = \left(63 \times 0.95^3 + 0.70 \times Q_i\right) \times \frac{0.95^0+0.95^1+\cdots+0.95^{11}}{12}$, where the average number of analyst demand is 90, 63 analysts in April and a retention rate is 0.95 consistently. See also the first hidden assumption.

Hiring on July 1 is assumed here. Let Q_i be the number of offers to make in month i, i.e., if it is not a month the company recruits, then $Q_i = 0$. In month i, if the bank makes offers to graduating students, then they know that they only expect to have 70% of them accept, which is fixed. Therefore, let A_i the number of analysts who accept an offer to start in a month and assume $A_i \overset{\text{i.i.d.}}{\sim} \text{Binom}(Q_i, 0.7)$ after getting Q_i by equation (2). Note that we assume the average annual retention rate is always 0.95 because it doesn't seem to depend on the value in the past years but depends on other factors such as the depression and world trends. Eventually, we can calculate the number of analysts employed at the start of the month, denoted by $P_i \in \{P_i\}_{i=1}^T \equiv P$, as follows.

$$P_i = P_{i-1}R_{i-1} + A_i; \quad P_0 = 63 \tag{3}$$

If new graduates can choose when they start to work, i.e., in the scenario with the idea (ii), we also need to consider another acceptance rate of those who choose to delay their start date because some of them would change their mind before joining the company. Since Susan tells us it is usually between 70% to 100%, we assume the rate follows a uniform distribution of that range.

Furthermore, we calculate the monthly earnings E_i in month i and then the annual earnings. We suppose the fiscal year starts in July, following the customs in Australia. The monthly earning depends on whether there is a shortage (i.e., $P_i \leq D_i$) or an excess (i.e., $P_i \geq D_i$) of analysts in month i. After considering all types of revenue and cost, E_i can be calculated in the following way.

$$E_i = \begin{cases} \text{(4-1)}\ \ 3600P_i + 400D_i & \text{if } P_i \leq D_i \\ \text{(4-2)}\ \ 4000P_i = 4000D_i & \text{if } P_i = D_i \\ \text{(4-3)}\ \ -6000P_i + 10000D_i & \text{if } P_i \geq D_i \end{cases} \tag{4}$$

Coloured part in Figure 1 shows the relationship between the three equations. Observer that the more closed P_i get to the equilibrium, i.e., (4-2), the higher E_i we can expect.

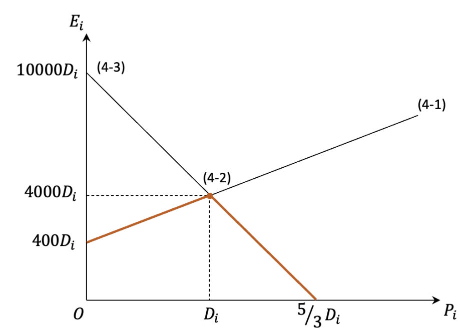


Figure 1. Monthly Earnings depending on Demand-supply Balance

Before moving on to the next section, we clarify some hidden assumptions. The client is supposed to estimate how many offers they should make *three months* before it indeed hires recruits; that is, the HR calculates equation (2) in April to recruit new graduates on 1 July. Also, while the hiring activity associates costs such as advertising on the job market and managing the documents for the new recruits, we *don't* consider it. This means that the company doesn't care about the cost even after the number of hiring per year gets four times. Moreover, we don't take it into account that new graduates are probably less productive soon after they join the company, rather they can work as senior employees do.

2.2.2 Four Scenarios We Consider Here, we clarify four scenarios discussed in the remaining part. For simplicity, we refer to the option of when new graduates join the workforce as the 'option'.

Scenario 1 hire new graduates only in *July*
Scenario 2 hire new graduates in July with the option – in either *July* or *September*)
Scenario 3 hire new graduates in *July* and *January*
Scenario 4 hire new graduates in July with the option – in either *July* or *September* – and January with the option – in either *January* or *March*

Figure 2 summarises the four scenarios in the calendar starting from April.

Figure 2. Annual Schedule on hiring new graduates

2.2.3 Simulation method We now demonstrate how we built the simulation we ran through in this analysis. A flowchart in Figure 3 is one of the most intuitive methods for understanding the simulation. We iterated this simulation 147 times for each scenario, i.e., $T = 147$. This number is equivalent to 12 years and 3 months. We chose this number because the long-term management plan in the company is usually around ten years. We also added three more months. Otherwise, we miss out on one value of annual earnings, which is less desired for a more informative analysis.

We have to mention several technical limitations and the way to escape them. When we implement the simulation, the flow in month i is divided into two parts: (a) getting D and (b) getting P, although the flowchart assumes all processes in each month are sequential. See appendix on page 8-20 to know the `R` syntax used in our simulation analysis. It is because getting the same D in all four scenarios is impossible otherwise, even though we set a random seed. Also note that we start the simulation from $i = 1$ instead of $i = 0$ due to the technical limitation, which results in the looking-forward code. The appearance of data frame we can obtain after executing code is displayed in Figure 4 and the definition of each variable in the column which is not mentioned so far are explained in Table 3.

Table 3. Description of variables in the column of Figure 4

variable	definition
`month`	actual name of month i
`Q_list`	the number of new graduates the bank tries to hire (all 0 except for the specific month to hire them)
`A`	the number of fresh recruits
`PminusD`	the number of excess supply (if positive)
`Annual_Ei_JUNE`	annual earnings summing from last July to this June (all 0 except for June)

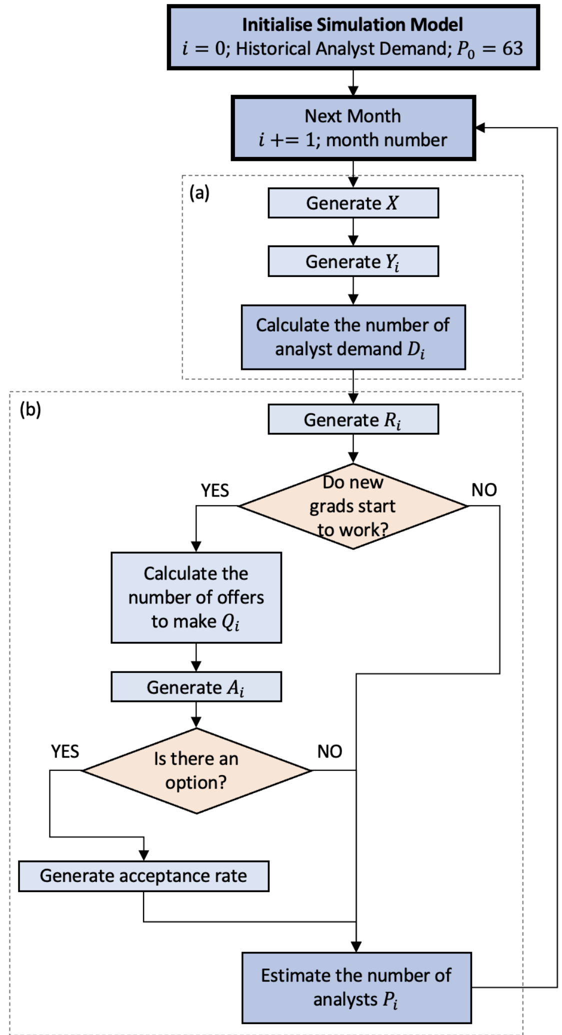
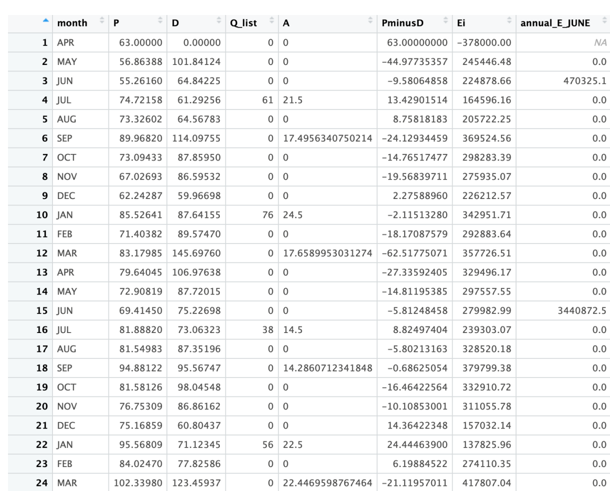


Figure 3. Flowchart of our simulation



	month	P	D	Q_list	A	PminusD	Ei	annual_E_JUNE
1	APR	63.00000	0.00000	0	0	63.00000000	-378000.00	NA
2	MAY	56.86388	101.84124	0	0	-44.97735357	245446.48	0.0
3	JUN	55.26160	64.84225	0	0	-9.58064858	224878.66	470325.1
4	JUL	74.72158	61.29256	61	21.5	13.42901514	164596.16	0.0
5	AUG	73.32602	64.56783	0	0	8.75818183	205722.25	0.0
6	SEP	89.96820	114.09755	0	17.4956340750214	-24.12934459	369524.56	0.0
7	OCT	73.09433	87.85950	0	0	-14.76517477	298283.39	0.0
8	NOV	67.02693	86.59532	0	0	-19.56839711	275935.07	0.0
9	DEC	62.24287	59.96698	0	0	2.27588960	226212.57	0.0
10	JAN	85.52641	87.64155	76	24.5	-2.11513280	342951.71	0.0
11	FEB	71.40382	89.57470	0	0	-18.17087579	292883.64	0.0
12	MAR	83.17985	145.69760	0	17.6589953031274	-62.51775071	357726.51	0.0
13	APR	79.64045	106.97638	0	0	-27.33592405	329496.17	0.0
14	MAY	72.90819	87.72015	0	0	-14.81195385	297557.55	0.0
15	JUN	69.41450	75.22698	0	0	-5.81248458	279982.99	3440872.5
16	JUL	81.88820	73.06323	38	14.5	8.82497404	239303.07	0.0
17	AUG	81.54983	87.35196	0	0	-5.80213163	328520.18	0.0
18	SEP	94.88122	95.56747	0	14.2860712341848	-0.68625054	379799.38	0.0
19	OCT	81.58126	98.04548	0	0	-16.46422564	332910.72	0.0
20	NOV	76.75309	86.86162	0	0	-10.10853001	311055.78	0.0
21	DEC	75.16859	60.80437	0	0	14.36422348	157032.14	0.0
22	JAN	95.56809	71.12345	56	22.5	24.44463900	137825.96	0.0
23	FEB	84.02470	77.82586	0	0	6.19884522	274110.35	0.0
24	MAR	102.33980	123.45937	0	22.4469598767464	-21.11957011	417807.04	0.0
25	APR	92.58378	101.76799	0	0	-9.18420285	374008.82	0.0

Figure 4. Output Image (Simulation for Scenario 4)

3 Results

3.1 Data Handling

Our analysis commences with the creation of a combined line chart illustrating the annual earnings across all four scenarios. Following this, we construct a matrix to evaluate the performance of each simulation model by categorising the variance between the number of analysts joining at the beginning of each month (P) and the required number of analysts (D) into three distinct classifications: Excess Demand (P<D), Excess Supply (P>D), and Equilibrium (P=D). Subsequently, we calculate the average annual recruitment statistics and the average annual earnings statistics over the simulation period for each scenario. Lastly, to gain deeper insights into the distribution of estimated earnings, we generate a boxplot for each scenario.

3.2 Results and Interpretation

Figure 5 depicts the anticipated yearly profits of Casterbridge Bank across a span of about 12 years, delineating four distinct scenarios. Notably, Scenario 4 consistently exhibits the highest annual earnings relative to the other scenarios, while Scenario 1 consistently registers the lowest annual profits. Additionally, there is considerable overlap in the annual earnings of Scenario 2 and Scenario 3, although they diverge notably in the initial years.

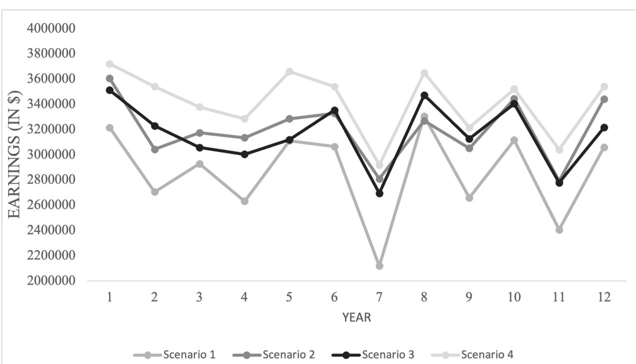


Figure 5. Annual Earnings across 4 Scenarios (in $) for a Period of 12 years

Table 4 provides an analysis of the performance of the four simulation scenarios. In Scenario 1, out of 147 months, the model indicates understaffing in 64 months and overstaffing in 78 months. Furthermore, the model accurately balances offers and analyst demand for only 5 months. In Scenario 2, out of 147 months, the model indicates understaffing in 78 months and overstaffing in 66 months. The model accurately balances offers and analyst demand for only 3 months. In Scenario 3, out of 147 months, the model indicates understaffing in 69 months and overstaffing in 78 months. However, it never accurately balances offers and analyst demand. In Scenario 4, out of 147 months, the model indicates understaffing in 81 months and overstaffing in 64 months. The model accurately balances offers and analyst demand for only 2 months.

As depicted in Table 5, Scenario 1 tends to have annually an overstaffing of 46 analysts, which correlates with estimated earnings of $2,676,096. Conversely, Scenario 2 consistently experiences an understaffing of 48 analysts annually, resulting in estimated earnings of $2,988,489. In Scenario 3, an overstaffing of 31 analysts annually yields estimated earnings amounting to $2,956,784.

Table 4. Simulation Model Performance Metrics

	Excess Demand ($P < D$)	Excess Supply ($P > D$)	Equilibrium ($P = D$)
Scenario 1	64	78	5
Scenario 2	78	66	3
Scenario 3	69	78	0
Scenario 4	81	64	2

Table 5. Forecast of Average Annual Earnings and Average Annual Recruitment

	Average Annual Excess Demand (per year)	Average Annual Earnings (in $)
Scenario 1	46	2,676,096
Scenario 2	-48	2,988,489
Scenario 3	31	2,956,748
Scenario 4	-29	3,190,804

Lastly, Scenario 4 exhibits an understaffing of 29 analysts annually, generating estimated earnings of $3,190,804. Within our model, earnings are shaped by both the firm's demand for analysts (D_i) and the analysts recruited by the firm at the beginning of the month (P_i). Consequently, when the firm overstaffs, monthly salary and indirect labour costs rise, leading to a decrease in earnings. Conversely, understaffing diminishes the monthly contributions to earnings from productive analysts, while also incurring costs associated with the bank transferring less efficient staff to meet shortages, further negatively impacting earnings. However, the expense of having an idle analyst outweighs the cost of transferring a less effective one. This suggests that it is more advantageous for the bank to be marginally understaffed rather than overstaffed.

Figure 6 shows the distribution of annual earnings across 4 scenarios. We can observe that, Scenario 1 consistently demonstrates median estimated earnings hovering around $2.9 million, with a relatively narrow interquartile range (IQR) spanning from approximately $2.7 million to $3.1 million. Similarly, Scenario 2 and Scenario 3 exhibit comparable median earnings of approximately $3.2 million, with IQRs spanning from $3.0 million to $3.4 million. Conversely, Scenario 4 stands out with higher median estimated earnings of around $3.5 million, alongside a broader IQR ranging from $3.3 million to $3.6 million. These findings suggest that while all scenarios maintain relatively stable earnings trajectories, Scenario 4 consistently outperforms the others, indicating a potentially more optimal staffing strategy.

4 Discussion

The results indicate that Scenario 4 consistently outperforms the other scenarios in terms of annual earnings across the 12-year period. The median annual earnings for Scenario 4 are around $3.5 million, with a broader IQR of $3.3 million to $3.6 million, suggesting that the staffing strategy employed in this scenario is the most optimal for maximizing the bank's earnings. Scenario 4 exhibits a higher frequency of understaffing (81 out of 147 months) compared to overstaffing (64 out of 147 months), with an average annual understaffing of 29 analysts. This moderate level of understaffing aligns with the observation that it is more advantageous

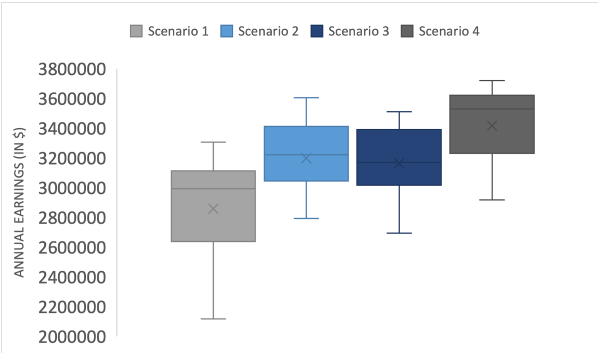


Figure 6. Distribution of Annual Earnings across Scenarios

for the bank to be marginally understaffed rather than overstaffed, as the cost of having idle analysts outweighs the cost of transferring less efficient staff during staff shortages.

Furthermore, the results reinforce the notion that moderate understaffing is preferable to overstaffing for maximizing earnings. Scenario 1, with an average annual overstaffing of 46 analysts, results in the lowest average annual earnings of $2,676,096, while Scenario 4, with an average annual understaffing of 29 analysts, yields the highest average annual earnings of $3,190,804. Scenarios 2 and 3 show similar median earnings but diverge in the initial years, suggesting that their short-term impact on earnings can vary, although their long-term results are comparable.

4.1 How many new analysts should the bank make offers to?

The result indicates that *Scenario 4 has the highest average annual earnings* of $3,190,814 despite having a slightly average annual understaffing of 29 analysts. This suggests that the number of offers made in Scenario 4 is likely the most optimal strategy. Based on the actual demand, the bank should maintain a slight understaffing and adjust its recruitment to hire fewer analysts than the demand indicates (about 29 analysts).

4.2 Did it make sense to offer recruits the option of starting at the beginning of September?

Scenario 1 involves starting in July, resulting in the lowest annual contributions among four scenarios over 12 years, with frequent overstaffing. This suggests that hiring new recruits once a year in July is not the most effective strategy for Casterbridge. Scenario 2 allows for starting in July or September. The simulation indicates the highest number of understaffing instances among the 4 scenarios, but the average earnings are the second highest. In comparison to scenario 1, scenario 2 offers more advantages due to its flexibility, which can help mitigate the overstaffing situation in scenario 1, but it still experiences a significant number of understaffing months.

In scenario 3, the bank offers new graduates the option to start in July or January. This scenario shows an average overstaffing situation with high average annual earnings of $2,957,648. Scenario 4 hires new graduates in July with the option to start in July or September, and in December with the option to start in January or March. Compared to scenario 3, scenario 4 generates higher annual earnings with a mild understaffing situation. Both scenarios provide multiple recruitment cycles, which could be advantageous in addressing fluctuating demand throughout the year, giving Casterbridge Bank the means to adjust their equilibrium of the firm's demand for analysts and the number of analysts employed.

While scenarios 1 and 3 demonstrate overstaffing, scenarios offering the option to start working in September show understaffing. In both cases, the outcomes show high revenue compared to scenarios 1 and 3. The company's demand for employees is highest in August and September, so having the option to join in September could reduce the excess demand situation in scenarios 1 and 3. Therefore, *it is advisable to offer new employees the option to start in September.*

4.3 Should the bank be more radical and hire twice each year instead of only once?

Hiring twice per year scenarios shows more profit than once. This approach could reduce the periods of both overstaffing and understaffing in all scenarios, especially considering the acceptance and retention rates.

4.4 Which strategy was best for the bank? How could we show our chairman the monetary implications of each strategy?

In our result, *Scenario 4 consistently outperforms in terms of annual earnings, median, and overall distribution.* This suggests that the strategy in scenario 4 is the most lucrative, while scenario 1 is the least profitable strategy. The result also cited that despite scenario 4 being slightly understaffed, the average annual earnings are the highest. In Table 1, scenario 4 has the highest number of understaffing months (81 out of 147). However, it still has the highest annual earnings, as shown in Figure 1. Therefore, this reinforces that scenario 4 is the most beneficial method. This scenario indicates that maintaining a lean staffing model with a slight deficit of analysts is more cost-effective and leads to better financial performance compared to other scenarios. To demonstrate the financial impact to the chairman, we can utilize a mix of visual and numerical representations. We will utilize Figures 1 and 2 to highlight the differences between scenarios, and Tables 1 and 2 to summarize the data. Analyzing each scenario and providing a summary of key recommendations will help the chairman to grasp each strategy easily. By presenting a comprehensive report with clear visual aids, detailed analyses, and actionable recommendations, the chairman can gain a thorough understanding of the financial implications of each strategy and make an informed decision regarding the optimal staffing approach for the Casterbridge Bank.

5 Other Consideration*

Why the simulation specification is (or is not) realistic for the business case?

The simulation model accurately reflects the dynamic nature of staffing needs and the complexities involved in managing analyst numbers. It takes into account the historical acceptance rates of job offers (70%) and the monthly retention rates, which vary between 80% and 100%. It addresses the cost of the recruitment process and the earnings of productive analysts that may hired. However, certain assumptions and limitations may not fully align with real-world scenarios.

Realistic Aspects

- Accounts for fluctuations in analyst demand throughout 12

years.
- Considers economic growth and its impact on analyst demand and new graduates.
- Incorporates random noise in demand, capturing inherent uncertainty.
- Accounts for analyst attrition rates, reflecting real-world employee turnover.
- Incorporates costs associated with over-staffing and under-staffing.

Potential Limitations

- A fixed 70% acceptance rate for job offers may not hold over an extended period.
- Does not consider external factors like competition or regulatory changes.
- Assumes static cost parameters (salaries, indirect costs) over the 12 years.
- Does not account for potential changes in the bank's business strategy or operational models.

While the simulation captures many essential elements, incorporating additional dynamic factors and updating assumptions periodically could further enhance its realism and applicability to real-world scenarios.

Are there other considerations such as inclusivity and sustainability that currently are not taken into account?
The current simulation does not explicitly address inclusivity and sustainability considerations, which are critical in modern business practices. The model does not account for diversity and inclusion in the hiring process and does not consider the potential impact of inclusivity initiatives on employee retention and engagement. Inclusivity can improve employee satisfaction, retention, and overall performance. The model could be enhanced by including metrics for diversity in hiring and retention rates among different demographic groups, ensuring that the staffing strategy supports an inclusive work environment Dudek (2023). The model also does not include environmental or social factors that may influence operations and staffing decisions, account for the potential impact of sustainable practices on operational costs, employee satisfaction, and reputation, and consider the long-term implications of staffing decisions on the bank's sustainability goals or commitments to social responsibility. Sustainability considerations might involve evaluating the environmental and social impact of staffing decisions. Adopting sustainable business practices, such as flexible working arrangements, could be incorporated into the simulation to reflect the bank's commitment to sustainability Shakeel et al. (2015). To address these considerations, the bank could incorporate the following into the simulation:

- Diversity and inclusion metrics related to the analyst workforce, and their potential impact on retention and productivity
- Environmental and social factors that may influence operations, stakeholder expectations, or reputational risks
- Potential cost savings or operational efficiencies associated with sustainable practices, and their impact on staffing requirements
- Long-term implications of staffing decisions on the bank's sustainability goals and commitments to social responsibility

By integrating these factors, the Casterbridge Bank can better align its staffing strategies with broader organisational goals related to inclusivity and sustainability, while also staying competitive in the industry.

6 Conclusion

The analysis conducted on Casterbridge Bank's staffing strategies over 12 years has yielded significant insights into optimising the bank's operations for sustained growth and efficiency. The report has elucidated the complexities of aligning staffing with fluctuating demand for analyst services to mitigate financial risks associated with over and under-staffing through meticulous examination and simulation modelling. Among the scenarios explored, Scenario 4 emerges as the most cost-effective approach. By offering recruits the option to start at different times of the year, the bank can achieve a balanced staffing model that minimises financial losses due to idle labour or inefficient staffing solutions. Furthermore, the analysis advocates hiring twice a year instead of just once to reduce over-staffing periods, thereby enhancing operational efficiency and financial performance. The findings emphasise the importance of maintaining a lean staffing model, with a slight deficit of analysts proving to be more advantageous than over-staffing. By presenting these conclusions along with detailed analyses and actionable recommendations, Casterbridge Bank can make informed decisions to refine its staffing approaches and ensure its continued success in a competitive financial landscape.

References

Dudek, Alicja (Dec. 2023). "How Does Inclusion Impact Employees in Organisations? Literature Review". In: *Acta Universitatis Lodziensis. Folia Oeconomica* 3, pp. 23–41. DOI: 10.18778/0208-6018.364.02.

Shakeel, Nausheen and Sahar But (2015). "Factors influencing employee retention: An integrated perspective". In: *Journal of Resources Development and Management* 6.1, pp. 32–49.

Errata

A part of the code used in this research article contains errors. Although X, a random variable that represents an economic growth rate per year, should be consistent for one year, we drew X from the certain probability distinction every "month." We apologise for this mistake and the fact that I couldn't fix it due to the time constraint. However, we can say it does not affect the implication at all because we use the same values for D between the four scenarios.

ETF5480: Group Project

Table of contents

Library

```r
library(MASS)
library(tidyverse)

# For Function `GeomSn` to Calculate a Summation of Geometric Series
library(DescTools)
```

Simulation Study

Scenario 1: Recruiting Once a Year in July

Initializing Variables

```r
simulation_time <- 12 * 12 + 3
P <- c(63, rep(0, simulation_time))
D <- c(0, rep(0, simulation_time))
month <- c("APR", numeric(simulation_time))
Q_list <- c(0, rep(0, simulation_time))
A <- c(0, rep(0, simulation_time))
```

Load Given Data

```r
# May-April
analyst_demand <- c(95, 75, 70, 70, 110, 105, 90, 65, 80, 90, 120, 105)
month_list <- c("JAN", "FEB", "MAR", "APR", "MAY", "JUN", "JUL", "AUG", "SEP", "OCT", "NOV", "DEC")
```

```r
## period of recruitment = annually
t <- 12
## when to recruit
recruit_month <- c(7)
```

Run Simulation

```r
# generate list of D
set.seed(23)
for (i in 1:simulation_time) {
  # index for the list
  index <- (i - 1) %% 12 + 1
  # month number (i.e., 1 = JAN, 2 = FEB, ..., 12 = DEC)
  month_number <- (i + 3) %% 12 + 1
  month[i + 1] <- month_list[c(month_number)]
  prev_month_number <- (i + 2) %% 12 + 1
  # random variables
  X <- rnorm(1, 0, 0.05)
  Yi <- rnorm(1, 0, 0.1)
  # get and calculate values
  Hi <- analyst_demand[index]
  Di <- Hi * (1 + X) * (1 + Yi)
  D[i + 1] <- Di
}
# generate list of P
j <- 1
for (i in 1:simulation_time) {
  # index for the list
  index <- (i - 1) %% 12 + 1
  # month number (i.e., 1 = JAN, 2 = FEB, ..., 12 = DEC)
  month_number <- (i + 3) %% 12 + 1
  prev_month_number <- (i + 2) %% 12 + 1
  # random variables
  if (prev_month_number %in% c(9, 1)) {
    R <- runif(1, 0.8, 1)
  } else if (prev_month_number %in% c(5, 6, 7, 8)) {
    R <- runif(1, 0.95, 1)
  } else {
    R <- runif(1, 0.9, 1)
  }
  # get and calculate values
  P_prev <- P[i]
  if (month_number %in% recruit_month) {
    # define Q
    #---cat("i-t+1=", i-t+1, "\n")
    Q <- ifelse(j == 1, 98, ((mean(D[(i - t + 1):i]) /
                             (GeomSn(1, 0.95, t - 1) / t)) - (P[(i + 1 - 3)] * (0.95**3))) / 0.70)
    Q_list[i + 1] <- ceiling(Q)
    # get Ai
```

3

```
    Ai <- rbinom(1, ceiling(Q), 0.70) # to get an integer for Q
    A[i + 1] <- Ai
    Pi <- P_prev * R + Ai
    j <- j + 1
  } else {
    Pi <- P_prev * R
    #--Q <- 0
  }
  P[i + 1] <- Pi
  #--cat("i:",i, "Q:", Q, "Ai:", Ai, "P_prev:", P_prev, "Pi:", Pi, "Di:", Di, "Hi:", Hi, "\n")
}
```

Store Results

```
scenario_1 <- data.frame(cbind(month, P, D, Q_list, A)) |>
  mutate(
    P = as.numeric(P),
    D = as.numeric(D),
    Q_list = as.numeric(Q_list)
  ) |>
  mutate(PminusD = P - D) |>
  mutate(Ei = ifelse(PminusD <= 0, 400 * D + 3600 * P, 10000 * D - 6000 * P))
sum_E_JUNE_scenario_1 <- c(NA, numeric(simulation_time))
for (i in 1:((simulation_time + 1) / 12 + 1)) {
  k <- 3 + (i - 1) * 12
  ifelse(k == 3, sum_E_JUNE_scenario_1[3] <- sum(scenario_1$Ei[2:3]),
    sum_E_JUNE_scenario_1[k] <- sum(scenario_1$Ei[(k - 11):(k)])
  )
}
scenario_1 <- scenario_1 |>
  mutate(annual_E_JUNE = sum_E_JUNE_scenario_1)
write_csv(scenario_1, "result/scenario_1_results.csv")
```

Clear Global Environment for Next Scenario

```
rm(list = (ls()[ls() != c("scenario_1")]))
```

Scenario 2: Recruiting Once a Year in July with an Option to Join in either in July or September

Initializing Variables

```
simulation_time <- 12 * 12 + 3
P <- c(63, rep(0, simulation_time))
D <- c(0, rep(0, simulation_time))
month <- c("APR", numeric(simulation_time))
Q_list <- c(0, rep(0, simulation_time))
A <- c(0, rep(0, simulation_time))
```

Load Given Data

```
# May-April
analyst_demand <- c(95, 75, 70, 70, 110, 105, 90, 65, 80, 90, 120, 105)
month_list <- c("JAN", "FEB", "MAR", "APR", "MAY", "JUN", "JUL", "AUG", "SEP", "OCT", "NOV", "DEC")
```

```
## period of recruitment = annually
t <- 12
## when to recruit
recruit_month <- c(7)
```

Run Simulation

```
# generate list of D
set.seed(23)
for (i in 1:simulation_time) {
  # index for the list
  index <- (i - 1) %% 12 + 1
  # month number (i.e., 1 = JAN, 2 = FEB, ..., 12 = DEC)
  month_number <- (i + 3) %% 12 + 1
  month[i + 1] <- month_list[c(month_number)]
  prev_month_number <- (i + 2) %% 12 + 1
  # random variables
  X <- rnorm(1, 0, 0.05)
  Yi <- rnorm(1, 0, 0.1)
  # get and calculate values
```

```
  Hi <- analyst_demand[index]
  Di <- Hi * (1 + X) * (1 + Yi)
  D[i + 1] <- Di
}
# generate list of P
j <- 1
for (i in 1:simulation_time) {
  # index for the list
  index <- (i - 1) %% 12 + 1
  # month number (i.e., 1 = JAN, 2 = FEB, ..., 12 = DEC)
  month_number <- (i + 3) %% 12 + 1
  #---cat("month:", month_number, "\n")
  prev_month_number <- (i + 2) %% 12 + 1
  # random variables
  if (prev_month_number %in% c(9, 1)) {
    R <- runif(1, 0.8, 1)
  } else if (prev_month_number %in% c(5, 6, 7, 8)) {
    R <- runif(1, 0.95, 1)
  } else {
    R <- runif(1, 0.9, 1)
  }
  # get and calculate values
  P_prev <- P[i]
  if (month_number %in% recruit_month) {
    #---cat("j=", j, "\n")
    #---cat("i-6-t+1=", i-6-t+1, "\n")
    # define Q
    Q <- ifelse(j == 1, 98, ((mean(D[(i - t + 1):i]) /
                            (GeomSn(1, 0.95, t - 1) / t)) - (P[(i + 1 - 3)] * (0.95**3))) / 0.70)
    Q_list[i + 1] <- ceiling(Q)
    # get Ai
    Ai <- rbinom(1, ceiling(Q), 0.70) # to get an integer for Q
    # half of them join the firm in July and the other half in September
    ## July(January) - half of them join the firm in this month
    A[i + 1] <- Ai / 2
    ## September(March) - the other half of them join the firm in this month with a certain acceptance rate
    acceptance_rate <- runif(1, 0.7, 1)
    if (i + 3 <= simulation_time) {
      A[i + 3] <- Ai / 2 * acceptance_rate
    }
    j <- j + 1
  }
```

6

```
  Pi <- P_prev * R + A[i + 1]
  P[i + 1] <- Pi
  #--cat("i:",i, "Q:", Q, "Ai:", Ai, "P_prev:", P_prev, "Pi:", Pi, "Di:", Di, "Hi:", Hi, "\n")
}
```

Store Results

```
## get the results
scenario_2 <- data.frame(cbind(month, P, D, Q_list, A)) |>
  mutate(
    P = as.numeric(P),
    D = as.numeric(D),
    Q_list = as.numeric(Q_list)
  ) |>
  mutate(PminusD = P - D) |>
  mutate(Ei = ifelse(PminusD <= 0, 400 * D + 3600 * P, 10000 * D - 6000 * P))
sum_E_JUNE_scenario_2 <- c(NA, numeric(simulation_time))
for (i in 1:((simulation_time + 1) / 12 + 1)) {
  k <- 3 + (i - 1) * 12
  ifelse(k == 3, sum_E_JUNE_scenario_2[3] <- sum(scenario_2$Ei[2:3]),
    sum_E_JUNE_scenario_2[k] <- sum(scenario_2$Ei[(k - 11):(k)])
  )
}
scenario_2 <- scenario_2 |>
  mutate(annual_E_JUNE = sum_E_JUNE_scenario_2)
write_csv(scenario_2, "result/scenario_2_results.csv")
```

Clear Global Environment for Next Scenario

```
rm(list = (ls()[ls() != c("scenario_1", "scenario_2")]))
```

Scenario 3: Recruiting Twice a Year in July and January

Initializing Variables

7

```r
simulation_time <- 12 * 12 + 3
P <- c(63, rep(0, simulation_time))
D <- c(0, rep(0, simulation_time))
month <- c("APR", numeric(simulation_time))
Q_list <- c(0, rep(0, simulation_time))
A <- c(0, rep(0, simulation_time))
```

Load Given Data

```r
# May-April
analyst_demand <- c(95, 75, 70, 70, 110, 105, 90, 65, 80, 90, 120, 105)
month_list <- c("JAN", "FEB", "MAR", "APR", "MAY", "JUN", "JUL", "AUG", "SEP", "OCT", "NOV", "DEC")
```

```r
## period of recruitment = annually
t <- 6
## when to recruit
recruit_month <- c(7, 1)
```

Run Simulation

```r
# generate list of D
set.seed(23)
for (i in 1:simulation_time) {
  # index for the list
  index <- (i - 1) %% 12 + 1
  # month number (i.e., 1 = JAN, 2 = FEB, ..., 12 = DEC)
  month_number <- (i + 3) %% 12 + 1
  month[i + 1] <- month_list[c(month_number)]
  prev_month_number <- (i + 2) %% 12 + 1
  # random variables
  X <- rnorm(1, 0, 0.05)
  Yi <- rnorm(1, 0, 0.1)
  # get and calculate values
  Hi <- analyst_demand[index]
  Di <- Hi * (1 + X) * (1 + Yi)
  D[i + 1] <- Di
}
j <- 1
for (i in 1:simulation_time) {
```

8

```r
# index for the list
index <- (i - 1) %% 12 + 1
# month number (i.e., 1 = JAN, 2 = FEB, ..., 12 = DEC)
month_number <- (i + 3) %% 12 + 1
#---cat("month:", month_number, "\n")
prev_month_number <- (i + 2) %% 12 + 1
# random variables
if (prev_month_number %in% c(9, 1)) {
  R <- runif(1, 0.8, 1)
} else if (prev_month_number %in% c(5, 6, 7, 8)) {
  R <- runif(1, 0.95, 1)
} else {
  R <- runif(1, 0.9, 1)
}
# get and calculate values
P_prev <- P[i]
if (month_number %in% recruit_month) {
  # define Q
  #---cat("i-t+1=", i-t+1, "\n")
  Q <- ifelse(j %in% c(1, 2),
    ifelse(j == 1,
      (((70 + 70 + 110 + 105 + 90 + 65) / 6) /
         (GeomSn(1, 0.95, 5) / 6) - 63 * (0.95**3)) / 0.7,
      # average demand of July-Dec in historical data
      (((80 + 90 + 120 + 105 + 95 + 75) / 6) /
         (GeomSn(1, 0.95, 5) / 6) - 63 * (0.95**3)) / 0.7),
    # average demand of Jan-June in historical data
    ((mean(D[(i + 1 - 6 - t + 1):(i + 1 - 6)]) /
        (GeomSn(1, 0.95, (t - 1)) / t)) - (P[(i + 1 - 3)] * (0.95**3))) / 0.7
  ) # average demand of July-Dec or Jan-June in simulated data    Q_list[i+1] <- ceiling(Q)
  Q_list[i + 1] <- ceiling(Q)
  # get Ai
  Ai <- rbinom(1, ceiling(Q), 0.70) # to get an integer for Q
  A[i + 1] <- Ai
  Pi <- P_prev * R + Ai
  j <- j + 1
} else {
  Pi <- P_prev * R
  #--Q <- 0
}
P[i + 1] <- Pi
#--cat("i:",i, "Q:", Q, "Ai:", Ai, "P_prev:", P_prev, "Pi:", Pi, "Di:", Di, "Hi:", Hi, "\n")
```

9

```
}
```

Store Results

```
scenario_3 <- data.frame(cbind(month, P, D, Q_list, A)) |>
  mutate(
    P = as.numeric(P),
    D = as.numeric(D),
    Q_list = as.numeric(Q_list)
  ) |>
  mutate(PminusD = P - D) |>
  mutate(Ei = ifelse(PminusD <= 0, 400 * D + 3600 * P, 10000 * D - 6000 * P))
sum_E_JUNE_scenario_3 <- c(NA, numeric(simulation_time))
for (i in 1:((simulation_time + 1) / 12 + 1)) {
  k <- 3 + (i - 1) * 12
  ifelse(k == 3, sum_E_JUNE_scenario_3[3] <- sum(scenario_3$Ei[2:3]),
    sum_E_JUNE_scenario_3[k] <- sum(scenario_3$Ei[(k - 11):k])
  )
}
scenario_3 <- scenario_3 |>
  mutate(annual_E_JUNE = sum_E_JUNE_scenario_3)
write_csv(scenario_3, "result/scenario_3_results.csv")
```

Clear Global Environment for Next Scenario

```
rm(list = (ls()[ls() != c("scenario_1", "scenario_2", "scenario_3")]))
```

Scenario 4: Recruiting Twice a Year in July and January with Option to Join either in July or September

Initializing Variables

```
simulation_time <- 12 * 12 + 3
P <- c(63, rep(0, simulation_time))
D <- c(0, rep(0, simulation_time))
month <- c("APR", numeric(simulation_time))
Q_list <- c(0, rep(0, simulation_time))
A <- c(0, rep(0, simulation_time))
```

10

Load Given Data

```
# May-April
analyst_demand <- c(95, 75, 70, 70, 110, 105, 90, 65, 80, 90, 120, 105)
month_list <- c("JAN", "FEB", "MAR", "APR", "MAY", "JUN", "JUL", "AUG", "SEP", "OCT", "NOV", "DEC")
```

```
## period of recruitment = annually
t <- 6
## when to recruit
recruit_month <- c(7, 1)
```

Run Simulation

```
# generate list of D
set.seed(23)
for (i in 1:simulation_time) {
  # index for the list
  index <- (i - 1) %% 12 + 1
  # month number (i.e., 1 = JAN, 2 = FEB, ..., 12 = DEC)
  month_number <- (i + 3) %% 12 + 1
  month[i + 1] <- month_list[c(month_number)]
  prev_month_number <- (i + 2) %% 12 + 1
  # random variables
  X <- rnorm(1, 0, 0.05)
  Yi <- rnorm(1, 0, 0.1)
  # get and calculate values
  Hi <- analyst_demand[index]
  Di <- Hi * (1 + X) * (1 + Yi)
  D[i + 1] <- Di
}
j <- 1
for (i in 1:simulation_time) {
  # index for the list
  index <- (i - 1) %% 12 + 1
  # month number (i.e., 1 = JAN, 2 = FEB, ..., 12 = DEC)
  month_number <- (i + 3) %% 12 + 1
  #---cat("month:", month_number, "\n")
  prev_month_number <- (i + 2) %% 12 + 1
  # random variables
  if (prev_month_number %in% c(9, 1)) {
```

```r
    R <- runif(1, 0.8, 1)
  } else if (prev_month_number %in% c(5, 6, 7, 8)) {
    R <- runif(1, 0.95, 1)
  } else {
    R <- runif(1, 0.9, 1)
  }
  # get and calculate values
  P_prev <- P[i]
  if (month_number %in% recruit_month) {
    #---cat("j=", j, "\n")
    #---cat("i-6-t+1=", i-6-t+1, "\n")
    # define Q
    Q <- ifelse(j %in% c(1, 2),
      ifelse(j == 1, (((70 + 70 + 110 + 105 + 90 + 65) / 6) /
                       (GeomSn(1, 0.95, 5) / 6) - 63 * (0.95**3)) / 0.7,
             # average demand of July-Dec in historical data
        (((80 + 90 + 120 + 105 + 95 + 75) / 6) /
           (GeomSn(1, 0.95, 5) / 6) - 63 * (0.95**3)) / 0.7
      ), # average demand of Jan-June in historical data
      ((mean(D[(i + 1 - 6 - t + 1):(i + 1 - 6)]) /
          (GeomSn(1, 0.95, (t - 1)) / t)) - (P[(i + 1 - 3)] * (0.95**3))) / 0.7
    ) # average demand of July-Dec or Jan-June in simulated data
    Q_list[i + 1] <- ceiling(Q)
    # get Ai
    Ai <- rbinom(1, ceiling(Q), 0.70) # to get an integer for Q
    # half of them join the firm in July and the other half in September
    ## July(January) - half of them join the firm in this month
    A[i + 1] <- Ai / 2
    ## September(March) - the other half of them join the firm in this month with a certain acceptance rate
    acceptance_rate <- runif(1, 0.7, 1)
    if (i + 3 <= simulation_time) {
      A[i + 3] <- Ai / 2 * acceptance_rate
    }
    j <- j + 1
  }
  Pi <- P_prev * R + A[i + 1]
  P[i + 1] <- Pi
  #--cat("i:",i, "Q:", Q, "Ai:", Ai, "P_prev:", P_prev, "Pi:", Pi, "Di:", Di, "Hi:", Hi, "\n")
}
```

12

Store Results

```r
## get the results
scenario_4 <- data.frame(cbind(month, P, D, Q_list, A)) |>
  mutate(
    P = as.numeric(P),
    D = as.numeric(D),
    Q_list = as.numeric(Q_list)
  ) |>
  mutate(PminusD = P - D) |>
  mutate(Ei = ifelse(PminusD <= 0, 400 * D + 3600 * P, 10000 * D - 6000 * P))
sum_E_JUNE_scenario_4 <- c(NA, numeric(simulation_time))
for (i in 1:((simulation_time + 1) / 12 + 1)) {
  k <- 3 + (i - 1) * 12
  ifelse(k == 3, sum_E_JUNE_scenario_4[3] <- sum(scenario_4$Ei[2:3]),
    sum_E_JUNE_scenario_4[k] <- sum(scenario_4$Ei[(k - 11):k])
  )
}
scenario_4 <- scenario_4 |>
  mutate(annual_E_JUNE = sum_E_JUNE_scenario_4)
write_csv(scenario_4, "result/scenario_4_results.csv")
```

Combine Annual Earnings From All the Simulation Models

```r
annual_earnings <- data.frame(cbind(scenario_1$annual_E_JUNE,
                                    scenario_2$annual_E_JUNE,
                                    scenario_3$annual_E_JUNE,
                                    scenario_4$annual_E_JUNE))
annual_earnings <- annual_earnings |> filter(X1 != 0)
annual_earnings <- cbind(seq(0, ((simulation_time - 3) / 12)), annual_earnings)
names(annual_earnings) <- c("Year", "Scenario 1", "Scenario 2", "Scenario 3", "Scenario 4")
write_csv(annual_earnings, "result/Combine_Annual_Earnings.csv")
```

13