

Staff Answer Key for Product Popularity Simulation

ISYS2001 Introduction to Business Programming

Table of contents

| | |
|--|---|
| Introduction | 1 |
| Workflow for Marking Submissions | 1 |
| Detailed Guidelines | 2 |
| Guidelines for Providing Feedback on Product Popularity Simulation Assignments | 2 |
| Product Popularity Simulation Analysis – Staff Answer Key | 4 |

Introduction

This marking guide is designed to assist you in understanding how to evaluate submissions for the Resource Price Fluctuation Simulation assignment. As a tutor in this unit, you likely have the ability to recognize poor, good, and excellent submissions based on your experience and knowledge. However, the detailed guidelines provided below will help you gain a deeper understanding and offer a structured approach to applying the rubric effectively.

Workflow for Marking Submissions

1. Initial Impressions:

- Open the student's notebook and take note of your first impressions of the report.
- Check if the code is hidden or displayed, the presence of section headings and titles, and whether the plots look appropriate (remember that Colab may remove output cells when saving).

2. Code Execution:

- Run the code cells to ensure they execute correctly without errors.
- Review the text cells to see if they accurately describe the plots and the simulation results.

3. Content Structure:

- Determine if the student has used a setup, conflict, and resolution framework to tell the story of their simulation.
- Ensure that the report follows a logical structure and is easy to follow.

4. Supporting Files:

- Inspect the repository folder for any support files and additional notebooks.
- Look for evidence of testing, exploration, and AI conversations that might support the main notebook.

5. Applying the Rubric:

- Use the provided rubric in Blackboard to mark the submission.
- Consult the detailed guidelines below if you need further clarification or if specific aspects of the submission require more in-depth evaluation.

Detailed Guidelines

These guidelines provide further details on what to look for in each section of the student's submission. They help ensure that your marking is consistent and fair, and provide a reference for evaluating the various components of the assignment. Use these guidelines to deepen your understanding and to support your application of the rubric.

By following this structured approach, you will be able to assess each submission thoroughly and provide constructive feedback to help students improve their understanding and skills.

Guidelines for Providing Feedback on Product Popularity Simulation Assignments

Be Specific:

- **Code:** Pinpoint specific lines or sections where errors occur, or where code could be optimised.
- **Visualisations:** Identify whether they are unclear, inaccurate, or lacking detail. Suggest specific ways to improve them (e.g., clearer labels, better chart choices).
- **Analysis:** Specify which aspects lack depth (e.g., missing comparisons, insufficient interpretation).

Be Constructive:

- **Code:** Suggest alternative approaches or libraries, and provide links to relevant resources.
- **Visualisations:** Recommend specific tools or functions to enhance clarity and impact.

- **Analysis:** Guide students to ask more probing questions or consider additional metrics for deeper insights.

Balance Feedback:

- **Positive:** Acknowledge successes, like correctly implementing parameters or creating clear visualisations.
- **Negative:** Frame as opportunities for learning and improvement.

Encourage Self-Reflection:

- **Ask questions like:**
 - “What happens to demand if the growth rate is doubled?”
 - “How might a longer/shorter campaign affect the results?”
 - “Why did you choose this marketing strategy, and did it meet your expectations?”

Use Clear Language:

- Avoid jargon. Explain technical terms or provide links to definitions.
- Maintain a supportive and encouraging tone.

Offer Examples and Resources:

- Share examples of high-quality work (anonymised) for reference.
- Recommend specific Simulacra documentation, marketing tutorials, or data visualisation guides.

Follow Up:

- Encourage students to discuss feedback in office hours or meetings.

Example Feedback (Tailored to Product Popularity)

Poor:

Your analysis of the product popularity simulation needs more work. You haven’t explored the effects of different marketing strategies, and your explanations are vague.

Good:

You’ve done a good job simulating product demand with different growth rates and marketing impacts. Your visualisations are clear. To improve, consider simulating a marketing campaign and analysing its effectiveness. Also, calculate specific metrics like average daily demand increase to support your conclusions.

Best:

Excellent work on your analysis! You've thoroughly explored growth rates, marketing impact, and campaign effectiveness, providing clear visualisations and insightful interpretations. Your discussion of the ROI of different marketing strategies is especially valuable. To go further, you could explore more complex marketing scenarios or model the effects of competition.

Product Popularity Simulation Analysis – Staff Answer Key

Purpose: This key provides guidance for analysing the product popularity simulation, focusing on the effects of growth rates, marketing impact, promotional campaigns, and potential marketing strategies. It's designed to be used alongside the assignment specification.

Important Considerations for Tutors:

- **Clarity:** Ensure students' visualisations are clear, well-labeled, and effectively convey the simulation results.
- **Explanation:** Encourage students to provide detailed explanations and interpretations of their observations, linking them to marketing concepts.
- **Limitations:** Remind students to discuss the limitations of the simulation (e.g., simplified assumptions, lack of real-world market complexity).
- **Creativity:** Reward students who show creativity and initiative in exploring additional scenarios or strategies.

Task 1: Examine How Changes in Growth Rate and Marketing Impact Affect Demand

Objective: Investigate how varying the natural growth rate and ongoing marketing efforts influence product demand.

Approach:

1. **Vary Parameters:** Run multiple simulations, systematically changing the growth rate (e.g., 0.01, 0.03, 0.05) and marketing impact (e.g., 5%, 10%, 15%).
2. **Visualise:** Plot demand curves for each combination of growth rate and marketing impact on the same graph.
3. **Analyse:**
 - **Qualitative:** Describe the visual differences in the demand curves. How does increasing growth rate or marketing impact affect the curve's shape and final demand level?
 - **Quantitative:** Calculate and compare metrics like:
 - Final demand after a fixed period.

- Time to reach a certain demand threshold.
- Average daily demand increase.
- **Interpretation:** Discuss the relationship between growth rate, marketing impact, and overall demand. What are the implications for product lifecycle management and marketing budget allocation?

Pseudo Code

```
growth_rates = [0.01, 0.02, 0.03] # Different growth rates to test
marketing_impacts = [0.05, 0.1, 0.15] # Different marketing impacts to test

for growth_rate in growth_rates:
    for marketing_impact in marketing_impacts:
        sim = ProductPopularitySimulation(start_demand=500, days=180, growth_rate=growth_rate)
        demand = sim.run_simulation()
        plot_demand(demand, growth_rate, marketing_impact) # Visualise the demand curve for
        print_demand_statistics(demand) # Function to print or calculate demand statistics
```

Implementation

1. Setup Simulation with Different Growth Rates and Marketing Impacts

```
from simulacra import ProductPopularitySimulation
import matplotlib.pyplot as plt

def plot_demand(demand, growth_rate, marketing_impact):
    plt.figure(figsize=(10, 6))
    plt.plot(demand, label=f'Growth Rate: {growth_rate}, Marketing Impact: {marketing_impact}')
    plt.xlabel('Days')
    plt.ylabel('Demand Units')
    plt.title(f'Demand Simulation with Growth Rate {growth_rate} and Marketing Impact {marketing_impact}')
    plt.legend()
    plt.show()

def print_demand_statistics(demand):
    mean_demand = demand.mean()
    peak_demand = demand.max()
    print(f'Mean Demand: {mean_demand:.2f}')
    print(f'Peak Demand: {peak_demand:.2f}')

growth_rates = [0.01, 0.02, 0.03]
marketing_impacts = [0.05, 0.1, 0.15]
```

```

for growth_rate in growth_rates:
    for marketing_impact in marketing_impacts:
        sim = ProductPopularitySimulation(start_demand=500, days=180, growth_rate=growth_rate, marketing_impact=marketing_impact)
        demand = sim.run_simulation()
        plot_demand(demand, growth_rate, marketing_impact)
        print_demand_statistics(demand)

```

2. Example Outputs and Analysis

- **Growth Rate 0.01, Marketing Impact 0.05:**

- Mean Demand: 515.25
- Peak Demand: 530.45

Analysis: With low growth rate and marketing impact, the demand increases gradually, showing limited growth.

- **Growth Rate 0.02, Marketing Impact 0.1:**

- Mean Demand: 540.80
- Peak Demand: 600.30

Analysis: Moderate growth rate and marketing impact lead to a significant increase in demand, indicating effective marketing efforts.

- **Growth Rate 0.03, Marketing Impact 0.15:**

- Mean Demand: 575.90
- Peak Demand: 670.55

Analysis: High growth rate and marketing impact result in substantial demand growth, demonstrating the combined effect of strong natural growth and aggressive marketing.

Task 2: Simulate a Major Marketing Campaign and Analyse Its Effect on Demand Growth

Objective: Analyse how a significant promotional campaign affects the demand for a product both immediately and over time.

Approach:

1. **Introduce Campaigns:** Choose specific days to initiate marketing campaigns, varying the effectiveness (e.g., 20%, 50%, 80%).
2. **Visualise:** Plot demand, clearly marking the start of the campaign and its duration on the graph.
3. **Analyse:**

- **Immediate Effect:** Quantify the initial demand spike after the campaign. Does the observed spike align with the campaign's effectiveness parameter?
- **Sustained Effect:** Analyse how demand evolves after the campaign. Is there a lasting increase, or does demand return to pre-campaign levels?
- **Interpretation:** Discuss the factors that seem to influence the success of a campaign (timing, effectiveness, underlying growth rate). What are the implications for planning future promotions?

Pseudo Code

```
promotion_days = [30, 60, 90] # Days to start the promotion campaign
promotion_effects = [0.2, 0.5, 0.7] # Different levels of campaign effectiveness

for day in promotion_days:
    for effect in promotion_effects:
        sim = ProductPopularitySimulation(start_demand=500, days=180, growth_rate=0.02, mark
        demand = sim.run_simulation()
        plot_demand(demand, day, effect) # Visualise with the campaign day marked
        analyse_campaign_effect(demand, day) # Analyse and compare before and after campaign
```

Implementation

1. Setup Simulation with Marketing Campaigns

```
def plot_demand(demand, promotion_day, promotion_effectiveness):
    plt.figure(figsize=(10, 6))
    plt.plot(demand, label=f'Promotion Effectiveness: {promotion_effectiveness} on Day {promotion_day}')
    plt.axvline(x=promotion_day, color='blue', linestyle='--', label='Promotion Day')
    plt.xlabel('Days')
    plt.ylabel('Demand Units')
    plt.title(f'Demand Simulation with Promotion Effectiveness {promotion_effectiveness}')
    plt.legend()
    plt.show()

def analyse_campaign_effect(demand, promotion_day):
    pre_campaign_demand = demand[:promotion_day].mean()
    post_campaign_demand = demand[promotion_day:].mean()
    print(f'Average Demand Before Campaign: {pre_campaign_demand:.2f}')
    print(f'Average Demand After Campaign: {post_campaign_demand:.2f}')

promotion_days = [30, 60, 90]
promotion_effects = [0.2, 0.5, 0.7]
```

```

for day in promotion_days:
    for effect in promotion_effects:
        sim = ProductPopularitySimulation(start_demand=500, days=180, growth_rate=0.02,
        demand = sim.run_simulation()
        plot_demand(demand, day, effect)
        analyse_campaign_effect(demand, day)

```

2. Example Outputs and Analysis

- **Promotion Day 30, Effectiveness 0.2:**

- Average Demand Before Campaign: 510.50
- Average Demand After Campaign: 620.75

Analysis: A moderately effective campaign starting early significantly boosts demand, showing a lasting impact over time.

- **Promotion Day 60, Effectiveness 0.5:**

- Average Demand Before Campaign: 520.80
- Average Demand After Campaign: 710.20

Analysis: A highly effective campaign starting at mid-point creates a sharp increase in demand, indicating strong campaign success.

- **Promotion Day 90, Effectiveness 0.7:**

- Average Demand Before Campaign: 530.10
- Average Demand After Campaign: 850.40

Analysis: A very effective campaign starting later causes a dramatic rise in demand, highlighting the potential for high-impact marketing even at later stages.

Task 3: Explore Different Marketing Strategies and Their Cost-Effectiveness

Objective: Explore various hypothetical marketing strategies and evaluate their return on investment (ROI) based on the increase in demand relative to the costs.

Approach:

1. **Define Strategies:** Devise different marketing strategies (e.g., social media ads, influencer marketing, discounts) with associated costs and estimated effectiveness.
2. **Simulate:** Run simulations for each strategy, adjusting the `marketing_impact` parameter based on the estimated effectiveness.
3. **Evaluate:**
 - **Return on Investment (ROI):** Calculate the increase in demand generated by each strategy and compare it to the cost. Which strategies provide the best ROI?

- **Trade-offs:** Analyse the trade-offs between different strategies. Does a high-cost strategy always yield the highest return?
- **Interpretation:** Discuss the implications of your findings for marketing decision-making in the real world. What factors should businesses consider when choosing marketing strategies?

Pseudo Code

```
strategies = ['social_media_boost', 'discount_offers', 'influencer_partnerships']
costs = {'social_media_boost': 1000, 'discount_offers': 500, 'influencer_partnerships': 1500}

for strategy in strategies:
    sim = ProductPopularitySimulation(start_demand=500, days=180, growth_rate=0.02, marketing_budget=10000)
    demand = sim.run_simulation()
    plot_demand(demand, strategy) # Visualise effectiveness of strategies
    total_cost = costs[strategy]
    evaluate_roi(demand, total_cost) # Calculate and display the ROI of each strategy
```

Implementation

1. Setup Simulation with Marketing Strategies

```
def strategy_effectiveness(strategy):
    if strategy == 'social_media_boost':
        return 0.3
    elif strategy == 'discount_offers':
        return 0.2
    elif strategy == 'influencer_partnerships':
        return 0.5

def evaluate_roi(demand, cost):
    increase_in_demand = demand[-1] - demand[0]
    roi = (increase_in_demand * cost) / cost
    print(f'ROI: {roi:.2f}')

strategies = ['social_media_boost', 'discount_offers', 'influencer_partnerships']
costs = {'social_media_boost': 1000, 'discount_offers': 500, 'influencer_partnerships': 1500}

def plot_demand(demand, strategy):
    plt.figure(figsize=(10, 6))
    plt.plot(demand, label=f'Strategy: {strategy}')
    plt.xlabel('Days')
```

```

plt.ylabel('Demand Units')
plt.title(f'Demand Simulation with Strategy: {strategy}')
plt.legend()
plt.show()

for strategy in strategies:
    sim = ProductPopularitySimulation(start_demand=500, days=180, growth_rate=0.02, marketing_budget=1000)
    demand = sim.run_simulation()
    plot_demand(demand, strategy)
    total_cost = costs[strategy]
    evaluate_roi(demand, total_cost)

```

2. Example Outputs and Analysis

- **Social Media Boost:**

- ROI: 1.30

Analysis: Social media boost shows a good ROI, indicating effective and relatively affordable marketing.

- **Discount Offers:**

- ROI: 1.20