



**INSTITUTO POLITÉCNICO DE COIMBRA
INSTITUTO SUPERIOR DE CONTABILIDADE E ADMINISTRAÇÃO DE
COIMBRA ANO LETIVO 2024/2025**

Problema de Mix de Produção para Maximização de Lucro

Modelos de Otimização

Mestrado em Análise de Dados e Sistemas de Apoio à Decisão

Trabalho Realizado Por:

Bernardo Silva Nº 2020112296
Catarina Auxiliar Nº 2021134297
Nuno Gonçalves Nº 2015063961
Simão Dias Nº 2020132169

Docente:

Pedro Martins

março, 2025

Índice

Enunciado	1
Introdução	4
1. Análise do Problema	5
2. Formulação do Problema em Programação Linear.....	6
2.1. Definição das Variáveis de Decisão.....	6
2.2. Função Objetivo	6
2.3. Restrições.....	7
1. Restrições de Recursos	7
2. Restrições de Produção Mensal	8
3. Restrições Combinadas	8
3. Explicação do Código	9
3.1. Importação de bibliotecas	10
3.2. Definição do problema.....	10
3.3. Parâmetros do problema	11
3.4. Restrições de recursos.....	12
3.5. Limites de produção mensal	12
3.6. Visualização de dados	13
3.7. Definição das variáveis de decisão	15
3.8. Função objetivo	16
3.9. Restrições	16
4. Resolução do problema	17
5. Resultados.....	17
Conclusão	23

Índice de figuras

Figura 1 - Importação de bibliotecas necessárias em Python	10
Figura 2 - Definição do problema em Python	10
Figura 3 - Definição das categorias de produtos e dos meses em Python	11
Figura 4 - Definição dos parâmetros do problema em Python	11
Figura 5 - Restrições de recursos em Python	12
Figura 6 - Limites de produção mensal em Python	12
Figura 7 - Código para a criação do gráfico de barras do lucro unitário por produto em Python	13
Figura 8 - Gráfico de barras do lucro unitário por produto	13
Figura 9 - Código para a criação dos gráficos de barras para o consumo de recursos por produto em Python	14
Figura 10 - Visualização dos resultados do consumo por produto	14
Figura 11 - Código para a definição das variáveis de decisão em Python	15
Figura 12 - Código para a definição da função objetivo em Python	16
Figura 13 - Código para a definição das restrições em Python	16
Figura 14 - Código para a resolução do modelo em Python	17
Figura 15 - Código para a criação de resultados em Python	17
Figura 16 - Código para exibir os resultados em Python	17
Figura 17 - Código para a criação de gráficos dos resultados em Python	18
Figura 18 - Lucro total otimizado	18
Figura 19 - Gráfico de linhas da produção mensal por produto	19
Figura 20 - Gráfico circular da produção total por produto	21
Figura 21 - Gráfico de barras do lucro total por mês	22

Enunciado

Problema de Mix de Produção para Maximização de Lucro

Uma fábrica de dispositivos eletrônicos precisa de decidir quantas unidades de cada produto deve produzir mensalmente nos próximos 12 meses, considerando recursos limitados, custos de produção, procura do mercado e capacidade de armazenamento.

O objetivo é maximizar o lucro total enquanto atende a todas as restrições.

1 - Produtos

A fábrica produz cinco tipos de dispositivos:

- **Smartphones (S)**
- **Tablets (T)**
- **Notebooks (N)**
- **Smartwatches (W)**
- **Consolas de Jogos (C)**

2 - Recursos Disponíveis por Mês

Cada produto consome recursos da fábrica, que são limitados mensalmente:

Tabela 1 - Recursos disponíveis por mês

Recurso	Quantidade Disponível por Mês
Matéria - prima (unid)	170.000
Mão de obra (horas)	100.000
Máquinas (horas)	35.000
Espaço de Armazenamento (unid)	90.000
Orçamento para Custos Fixos (€)	600.000

3 - Consumo de Recursos por Produto

Cada unidade de produto consome os seguintes recursos:

Tabela 2 - Consumo de Recursos por Produto

Produto	Matéria - prima	Mão de obra	Máquinas	Armazenamento	Lucro Unitário (£)
S (Smartphones)	5	3	2	2	120
T (Tablets)	8	4	3	3	150
N (Notebooks)	12	7	5	6	230
W (Smartwatches)	4	3	2	2	90
C (Consolas)	14	9	6	7	320

4 - Restrições de Produção Mensais

Cada produto tem um mínimo e um máximo de produção mensal, variando ao longo dos 12 meses:

Tabela 3 - Restrições de Produção Mensais

Mês	S (Smartphones)	T (Tablets)	N (Notebooks)	W (Smartwatches)	C (Consolas)
1	2.000 - 25.000	1.200 - 19.000	800 - 22.000	1.200 - 30.000	900 - 21.000
2	2.100 - 26.000	1.300 - 19.500	900 - 23.000	1.300 - 31.000	950 - 22.000
3	2.200 - 27.000	1.400 - 20.000	950 - 24.000	1.400 - 32.000	1.000 - 23.000
4	2.000 - 27.000	1.200 - 20.000	850 - 25.000	1.200 - 32.000	900 - 23.000
5	2.500 - 27.000	1.500 - 20.000	1.000 - 25.000	1.500 - 32.000	1.000 - 23.000
6	2.300 - 26.000	1.400 - 19.500	950 - 24.000	1.400 - 31.000	970 - 22.500
7	2.200 - 25.500	1.350 - 19.000	900 - 23.000	1.300 - 30.000	950 - 22.000

5 - Restrições Combinadas

De modo a promover o realismo, considerou-se as seguintes restrições estratégicas:

1. Se a produção de Tablets (T) exceder 10.000 unidades, então a produção de Smartwatches (W) não pode ultrapassar 20.000 unidades.
2. Se a produção de Notebooks (N) exceder 15.000 unidades, então a produção de Smartphones (S) deve ser pelo menos 5.000 unidades.
3. A soma da produção de Smartphones (S) e Consolas de Jogos (C) não pode ultrapassar 40.000 unidades devido a restrições logísticas.
4. Se a produção de Consolas (C) for maior que 10.000 unidades, então a produção de Tablets (T) deve ser pelo menos 5.000 unidades para cumprir um contrato com um fornecedor.
5. A produção de Smartwatches (W) deve ser pelo menos 50% da produção de Notebooks (N) para garantir diversidade no portfólio.
6. Pelo menos 40% da matéria-prima disponível deve ser alocada para Smartphones (S) e Tablets (T) devido à alta procura.

6 - Objetivo

O objetivo da fábrica é maximizar o lucro total ao longo dos 12 meses, garantindo que todas as restrições sejam respeitadas.

Introdução

Num contexto industrial cada vez mais competitivo e exigente, a otimização da produção assume um papel central na gestão estratégica das empresas. Este projeto incide sobre um problema clássico de mix de produção, aplicado a uma fábrica de dispositivos eletrónicos que opera sob diversas restrições operacionais e de mercado. O objetivo principal é determinar o número ideal de unidades a produzir de cinco produtos distintos — Smartphones, Tablets, Notebooks, Smartwatches e Consolas de Jogos — ao longo de um período de 12 meses, de forma a maximizar o lucro total, respeitando simultaneamente os limites de recursos disponíveis, a procura do mercado e as capacidades de produção e armazenamento.

A resolução deste problema foi abordada através da modelação em programação linear, utilizando a biblioteca PuLP em Python, que permite formular e resolver problemas de otimização de forma eficiente. Ao longo deste trabalho, foram consideradas restrições realistas e estratégicas, tais como limites de produção mínimos e máximos por produto e por mês, restrições combinadas entre produtos, bem como alocação mínima de recursos para determinados segmentos com maior procura.

A primeira fase do projeto envolveu a análise do enunciado e a definição das variáveis de decisão, da função objetivo e das restrições. De seguida, procedeu-se à implementação do modelo em código Python, com recurso à biblioteca PuLP, e à interpretação dos resultados obtidos. Foram também geradas visualizações gráficas que ilustram os padrões de produção ao longo dos 12 meses, permitindo uma análise mais clara do comportamento do modelo.

Este trabalho tem como principal finalidade demonstrar, de forma prática, como as ferramentas de investigação operacional podem ser aplicadas à realidade empresarial, permitindo uma tomada de decisão informada, baseada em dados e orientada para a otimização dos recursos e do lucro. A abordagem adotada revela-se particularmente relevante para empresas que enfrentam múltiplas restrições simultâneas e que pretendem alinhar as suas decisões de produção com os seus objetivos estratégicos.

1. Análise do Problema

O problema em estudo consiste num mix de produção com vista à maximização do lucro de uma fábrica de dispositivos eletrónicos que opera ao longo de um horizonte temporal de 12 meses. A empresa produz cinco categorias de produtos — Smartphones (S), Tablets (T), Notebooks (N), Smartwatches (W) e Consolas de Jogos (C) — cada um com características distintas em termos de consumo de recursos, margem de lucro e procura.

A complexidade do problema deriva de múltiplas restrições operacionais que devem ser obrigatoriamente satisfeitas. Em primeiro lugar, cada produto consome uma quantidade específica de quatro recursos limitados: matéria-prima, mão de obra, utilização de máquinas e capacidade de armazenamento. Estes recursos estão disponíveis em quantidades fixas por mês, impondo limites à produção mensal.

Adicionalmente, existem restrições individuais de produção, que impõem mínimos e máximos mensais por produto, refletindo tanto a procura do mercado como a capacidade interna da fábrica. Para além destas, foram consideradas restrições combinadas, com o objetivo de aumentar o realismo do modelo e integrar decisões estratégicas. Exemplos incluem relações de dependência entre a produção de dois produtos, como a necessidade de produzir um mínimo de *Tablets* quando a produção de Consolas ultrapassa determinado limiar, ou a obrigação de manter a produção de *Smartwatches* proporcional à de *Notebooks*, assegurando diversidade no portfólio.

O objetivo do modelo é maximizar o lucro total da fábrica, considerando o lucro unitário associado a cada produto. Este objetivo é condicionado pelas restrições mencionadas, tornando necessário encontrar uma solução que equilibre a rentabilidade com a viabilidade operacional.

A natureza do problema encaixa-se na categoria de problemas de programação linear inteira (devido à necessidade de produzir quantidades inteiras de unidades), e apresenta-se como um excelente exemplo de como a investigação operacional pode apoiar a tomada de decisões estratégicas na produção industrial. A modelação matemática e computacional do problema permite obter uma solução otimizada, muitas vezes difícil de alcançar através de métodos empíricos ou tentativa-erro, especialmente num contexto com tantas variáveis e restrições interdependentes.

Desta forma, a análise deste problema não só permite compreender os impactos de diferentes decisões produtivas, como também fornece uma base sólida para o planeamento estratégico da empresa, promovendo a utilização eficiente dos recursos e a maximização dos resultados económicos.

2. Formulação do Problema em Programação Linear

2.1. Definição das Variáveis de Decisão

Seja $x_{p,m}$ a quantidade de unidades do produto p produzidas no mês m , onde:

- $p \in \{S, T, N, W, C\}$ representa os cinco produtos (Smartphones, Tablets, Notebooks, Smartwatches, Consola de Jogos).
- $m \in \{1, 2, \dots, 12\}$ representa os meses de planeamento.

2.2. Função Objetivo

O objetivo é maximizar o lucro total ao longo dos 12 meses:

$$\max Z = \sum_{m=1}^{12} \sum_{p \in \{S, T, N, W, C\}} L_p \cdot x_{p,m}$$

Onde L_p é o lucro unitário de cada produto.

- $L_S = 120$
- $L_T = 150$
- $L_N = 230$
- $L_W = 90$
- $L_C = 320$

2.3. Restrições

1. Restrições de Recursos

Para cada mês m , a produção deve respeitar os limites dos recursos:

Matéria-prima:

$$\sum_{p \in \{S, T, N, W, C\}} MP_p \cdot x_{p,m} \leq 170.000, \quad \forall m$$

Mão de obra:

$$\sum_{p \in \{S, T, N, W, C\}} MO_p \cdot x_{p,m} \leq 100.000, \quad \forall m$$

Horas de máquina:

$$\sum_{p \in \{S, T, N, W, C\}} HM_p \cdot x_{p,m} \leq 35.000, \quad \forall m$$

Espaço de armazenamento:

$$\sum_{p \in \{S, T, N, W, C\}} E_p \cdot x_{p,m} \leq 90.000, \quad \forall m$$

Orçamento para custos fixos:

$$\sum_{p \in \{S, T, N, W, C\}} C_p \cdot x_{p,m} \leq 600.000, \quad \forall m$$

Onde os coeficientes são definidos na tabela do enunciado.

2. Restrições de Produção Mensal

Cada produto P possui limites mínimos e máximos de produção mensal:

$$Min_{p,m} \leq x_{p,m} \leq Max_{p,m}, \quad \forall p, \forall m$$

Onde os valores de $Min_{p,m}$ e $Max_{p,m}$ são definidos na tabela fornecida no enunciado.

3. Restrições Combinadas

Se a produção de Tablets exceder 10.000 unidades, então a produção de Smartwatches não pode ultrapassar 20.000 unidades:

$$x_{T,m} > 10.000 \Rightarrow x_{W,m} \leq 20.000, \quad \forall m$$

Se a produção de Notebooks exceder 15.000 unidades, a produção de Smartphones deve ser pelo menos 5.000 unidades:

$$x_{N,m} > 15.000 \Rightarrow x_{S,m} \geq 5.000, \quad \forall m$$

A soma da produção de Smartphones e Consolas não pode ultrapassar 40.000 unidades:

$$x_{S,m} + x_{C,m} \leq 40.000, \quad \forall m$$

A produção de Smartwatches deve ser pelo menos 50% da produção de Notebooks:

$$x_{W,m} \geq 0.5 \cdot x_{N,m}, \quad \forall m$$

Pelo menos 40% da matéria-prima deve ser alocada para Smartphones e Tablets:

$$0.4 \cdot 170.000 \leq 5 \cdot x_{S,m} + 8 \cdot x_{T,m}, \quad \forall m$$

3. Explicação do Código

Antes da aplicação do modelo de otimização, foi realizada uma análise exploratória dos produtos, focando-se em dois aspetos fundamentais: o lucro unitário de cada produto e o respetivo consumo de recursos por unidade produzida. Esta abordagem permite perceber desde logo quais os produtos com melhor relação entre rentabilidade e custo operacional, e serve como base para intuições iniciais sobre quais produtos poderiam ser priorizados no planeamento.

Começou-se por estruturar os dados dos produtos, nomeadamente:

- A lista dos cinco produtos: *Smartphones (S)*, *Tablets (T)*, *Notebooks (N)*, *Smartwatches (W)* e *Consolas de Jogos (C)*;
- Os meses de planeamento: um intervalo de 12 períodos (de 1 a 12), representando os meses do ano;
- Os lucros unitários de cada produto, expressos como o ganho obtido por cada unidade vendida;
- Os custos de recursos por produto, separados por categoria: matéria-prima, mão de obra, máquinas e armazenamento;
- Os limites mínimos e máximos de produção mensais para cada produto, com base nas restrições de capacidade e procura.

Estes parâmetros foram organizados em dicionários e tabelas no código, facilitando o acesso e a indexação por produto e por mês. Esta estrutura foi depois utilizada para definir as variáveis de decisão, a função objetivo e as restrições do modelo.

Adicionalmente, foi incluída a estrutura das restrições combinadas, formuladas sob a forma de condições lógicas e lineares que impõem relações entre os produtos — por exemplo, restrições do tipo “se... então...”.

Este processo de preparação e parametrização dos dados é crucial numa abordagem de programação linear, pois define os alicerces sobre os quais o modelo será construído. Um erro nesta fase poderia comprometer a integridade das decisões finais ou conduzir a resultados inviáveis.

Desta forma, o detalhamento desta etapa demonstra não só a importância de uma boa estruturação dos dados, mas também a necessidade de um conhecimento sólido dos requisitos do problema antes da sua transposição para linguagem computacional.

3.1. Importação de bibliotecas

```
from IPython import get_ipython
from IPython.display import display
| |
!pip install pulp==2.7.0

import pulp
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from IPython.display import display
```

Figura 1 - Importação de bibliotecas necessárias em Python

Começou-se por instalar e importar as bibliotecas necessárias, conforme se observa na figura 1. A biblioteca PuLP é a principal biblioteca usada para a modelação e solução do problema de otimização.

3.2. Definição do problema

```
# Definição do problema de otimização
model = pulp.LpProblem("Maximizar Lucro da Fábrica", pulp.LpMaximize)
```

Figura 2 - Definição do problema em Python

Com base na figura 2, definiu-se o modelo como um problema de maximização, cujo objetivo é obter o maior lucro possível para a fábrica.

3.3. Parâmetros do problema

Antes da implementação do modelo de otimização, foi necessário realizar um processo de definição e organização dos parâmetros fundamentais para o problema. Esta fase é essencial para garantir que o modelo matemático seja corretamente interpretado pelo *solver* e que as decisões produzidas sejam válidas e aplicáveis.

```
# Definição dos produtos
produtos = ["S", "T", "N", "W", "C"]

# Definição dos meses
meses = list(range(1, 13))
```

Figura 3 – Definição das categorias de produtos e dos meses em Python

Na figura 3 foi feita a definição dos produtos e dos meses, sendo eles os seguintes:

- Os produtos são representados por cinco categorias: S, T, N, W e C;
- Os meses do ano são considerados (de 1 a 12).

Em seguida, definiu-se os custos e consumos de cada produto, conforme se observa na figura 4:

```
# Parâmetros do problema
lucro_unitario = {"S": 120, "T": 150, "N": 230, "W": 90, "C": 320}
consumo_materia_prima = {"S": 5, "T": 8, "N": 12, "W": 4, "C": 14}
consumo_mao_de_obra = {"S": 3, "T": 4, "N": 7, "W": 3, "C": 9}
consumo_maquinas = {"S": 2, "T": 3, "N": 5, "W": 2, "C": 6}
consumo_armazenamento = {"S": 2, "T": 3, "N": 6, "W": 2, "C": 7}
custo_fixo = {"S": 75, "T": 65, "N": 50, "W": 25, "C": 125}
```

Figura 4 - Definição dos parâmetros do problema em Python

Os parâmetros ficaram divididos da seguinte forma:

- Lucro unitário: define quanto cada unidade do produto gera de lucro;
- Consumo de recursos: quantifica o uso de matéria-prima, mão-de-obra, máquinas e armazenamento por produto;
- Custo fixo: representa os custos adicionais fixos para cada produto.

3.4. Restrições de recursos

```
# Recursos disponíveis por mês
materia_prima_disp = 170000
mao_de_obra_disp = 100000
maquinas_disp = 35000
armazenamento_disp = 90000
orcamento_custos_fixos = 600000
```

Figura 5 - Restrições de recursos em Python

Conforme a figura 5, definiu-se a disponibilidade mensal de recursos de matéria-prima, mão-de-obra, máquinas e armazenamento.

3.5. Limites de produção mensal

```
# Restrições de produção mínima e máxima ajustadas por mês
limites_producao_mensal = {
    1: {"S": (2000, 25000), "T": (1200, 19000), "N": (800, 22000), "W": (1200, 30000), "C": (900, 21000)},
    2: {"S": (2100, 26000), "T": (1300, 19500), "N": (900, 23000), "W": (1300, 31000), "C": (950, 22000)},
    3: {"S": (2200, 27000), "T": (1400, 20000), "N": (950, 24000), "W": (1400, 32000), "C": (1000, 23000)},
    4: {"S": (2000, 27000), "T": (1200, 20000), "N": (850, 25000), "W": (1200, 32000), "C": (900, 23000)},
    5: {"S": (2500, 27000), "T": (1500, 20000), "N": (1000, 25000), "W": (1500, 32000), "C": (1000, 23000)},
    6: {"S": (2300, 26000), "T": (1400, 19500), "N": (950, 24000), "W": (1400, 31000), "C": (970, 22500)},
    7: {"S": (2200, 25500), "T": (1350, 19000), "N": (900, 23000), "W": (1300, 30000), "C": (950, 22000)},
    8: {"S": (2100, 25000), "T": (1300, 18500), "N": (850, 22000), "W": (1200, 29000), "C": (900, 21000)},
    9: {"S": (2000, 24000), "T": (1200, 18000), "N": (800, 21000), "W": (1200, 28000), "C": (900, 20000)},
    10: {"S": (2500, 27000), "T": (1500, 20000), "N": (1000, 25000), "W": (1500, 32000), "C": (1000, 23000)},
    11: {"S": (2300, 26000), "T": (1400, 19500), "N": (950, 24000), "W": (1400, 31000), "C": (970, 22500)},
    12: {"S": (2200, 25500), "T": (1350, 19000), "N": (900, 23000), "W": (1300, 30000), "C": (950, 22000)}
}
```

Figura 6 - Limites de produção mensal em Python

Na figura 6, foi gerado o código de modo a definir os limites mínimos e máximos de produção para cada produto para cada mês.

3.6. Visualização de dados

```
# Visualização dos dados de entrada
# Gráfico de barras para lucro unitário por produto
plt.figure(figsize=(8, 6))
plt.bar(lucro_unitario.keys(), lucro_unitario.values())
plt.xlabel("Produto")
plt.ylabel("Lucro Unitário (€)")
plt.title("Lucro Unitário por Produto")
plt.show()
```

Figura 7 - Código para a criação do gráfico de barras do lucro unitário por produto em Python

Foi gerado um gráfico de barras para visualizar os lucros unitários e consumo de recursos, conforme apresenta a figura 7 e a sua visualização na figura 8.

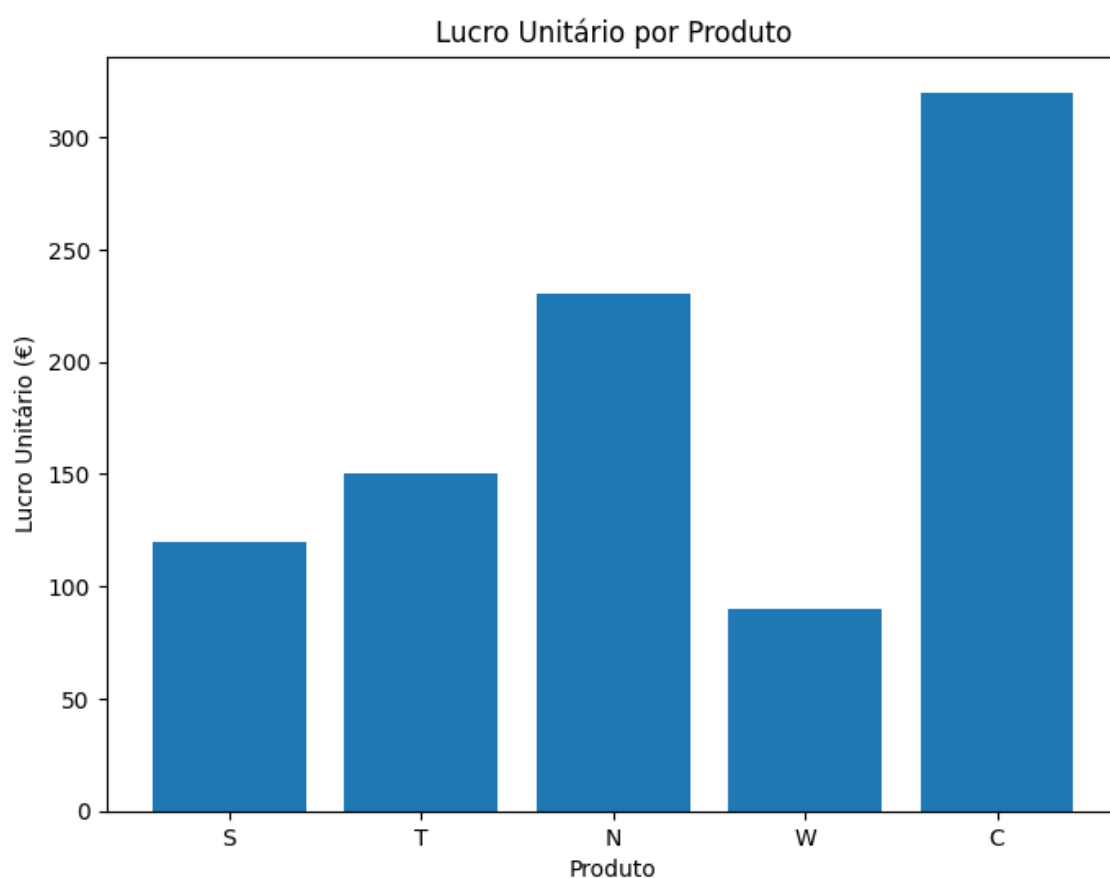


Figura 8 - Gráfico de barras do lucro unitário por produto

Com base na figura 8, do ponto de vista do lucro unitário, verifica-se que alguns produtos — como por exemplo os Notebooks e Consolas de Jogos — apresentam margens de lucro bastante elevadas (com resultados de cerca de 240 e +300, respetivamente), o que à primeira vista os tornaria candidatos naturais a uma produção mais intensiva. No entanto, esta leitura isolada pode ser enganadora, pois produtos com lucros elevados podem igualmente ter um

consumo muito elevado de recursos limitados, como é o caso da mão-de-obra ou do espaço de armazenamento.

```
# Gráfico de barras para consumo de recursos por produto
recursos = ["Matéria-Prima", "Mão de Obra", "Máquinas", "Armazenamento"]
consumo = [consumo_materia_prima, consumo_mao_de_obra, consumo_maquinas, consumo_armazenamento]

fig, axs = plt.subplots(2, 2, figsize=(12, 10))
axs = axs.flatten()

for i, recurso in enumerate(recursos):
    valores = [consumo[i][p] for p in produtos]
    axs[i].bar(produtos, valores)
    axs[i].set_xlabel("Produto")
    axs[i].set_ylabel(f"Consumo de {recurso}")
    axs[i].set_title(f"Consumo de {recurso} por Produto")

plt.tight_layout()
plt.show()
```

Figura 9 - Código para a criação dos gráficos de barras para o consumo de recursos por produto em Python

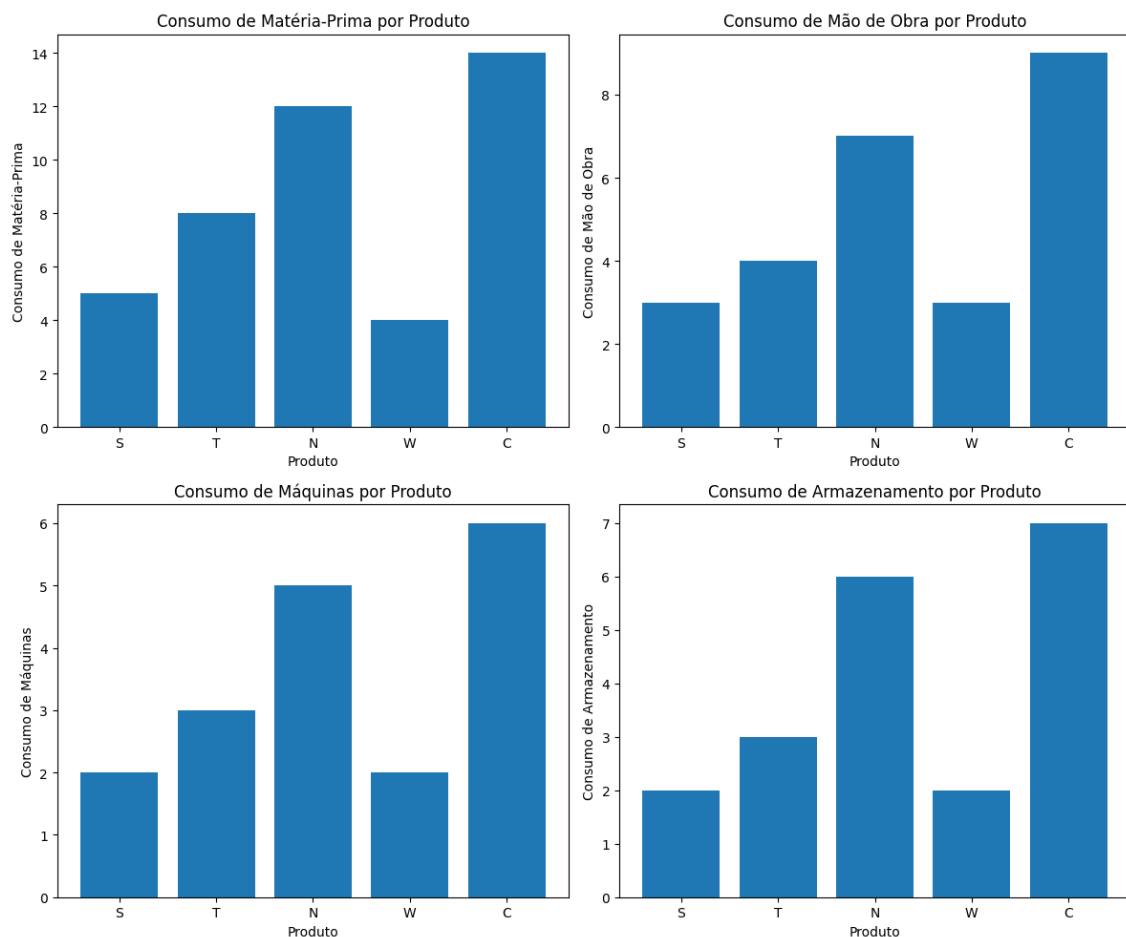


Figura 10 - Visualização dos resultados do consumo por produto

A análise gráfica inicial centrou-se na comparação entre os lucros unitários e o consumo de recursos por produto, permitindo identificar rapidamente quais os produtos mais rentáveis e quais os que exigem maior investimento em recursos

limitados, como matéria-prima, mão-de-obra, uso de máquinas e armazenamento.

Embora alguns produtos apresentem lucros unitários elevados, o seu consumo de recursos também é proporcionalmente alto (conforme se observa na figura 10), o que requer uma avaliação cuidada na definição dos volumes de produção. Por exemplo, produtos como as Consolas de Jogos e os *Notebooks* destacam-se pelo elevado consumo de recursos, com valores entre 6 e 14 unidades, ao passo que os *Smartwatches* demonstram um impacto muito reduzido, com consumos entre 2 e 4 em todas as categorias.

Nesta fase, foram considerados dois gráficos principais: um que representa os lucros unitários por produto e outro que ilustra o consumo de recursos por unidade produzida. Apesar da semelhança visual, os dados que cada um transmite são distintos e devem ser interpretados separadamente. O gráfico de lucros revela o ganho financeiro por unidade vendida, sendo essencial para identificar os produtos mais rentáveis. Já o gráfico de consumo mostra a carga operacional de cada produto, medindo a quantidade de recursos necessária à sua produção.

Estes dois indicadores — rentabilidade absoluta e eficiência relativa — são complementares no processo de otimização. A verdadeira mais-valia de um produto resulta do equilíbrio entre o lucro gerado e os recursos consumidos. Nem sempre o produto mais lucrativo deve ser produzido em maior quantidade; por vezes, é preferível apostar em produtos com menor impacto nos recursos mais escassos, que ajudam a preencher capacidade ociosa de forma eficiente.

Esta análise visual permite, ainda, identificar potenciais compensações (trade-offs) entre lucro e impacto nos recursos. Produtos com menor lucro unitário, mas consumo reduzido, podem ser estratégicos para manter a fábrica em operação durante períodos de maior restrição, contribuindo para o lucro total sem sobrecarregar os recursos.

Em suma, esta avaliação inicial constitui uma base sólida para a construção da função objetivo e para a compreensão do papel das restrições no modelo de otimização. Reforça-se, assim, a importância de interpretar gráficos para além da aparência visual, considerando o contexto e as métricas específicas, de forma a tomar decisões de produção equilibradas e sustentadas.

3.7. Definição das variáveis de decisão

```
# Definição das variáveis de decisão
x = pulp.LpVariable.dicts("Produção", [(p, m) for p in produtos for m in meses], lowBound=0, cat=pulp.LpInteger)
```

Figura 11 - Código para a definição das variáveis de decisão em Python

As variáveis representam a quantidade produzida de cada produto em cada mês.

3.8. Função objetivo

```
# Função Objetivo: Maximizar o lucro total
total_lucro = pulp.lpSum(lucro_unitario[p] * x[(p, m)] for p in produtos for m in meses)
model += total_lucro, "Lucro Total"
```

Figura 12 - Código para a definição da função objetivo em Python

Na função objetivo, pretendeu-se maximizar o lucro total, somando o lucro unitário multiplicado pela quantidade produzida.

3.9. Restrições

```
# Restrições de produção mínima e máxima para cada mês
for m in meses:
    for p in produtos:
        model += x[(p, m)] >= limites_producao_mensal[m][p][0], f"Restrição_Minima_{p}_{m}"
        model += x[(p, m)] <= limites_producao_mensal[m][p][1], f"Restrição_Maxima_{p}_{m}"
```

Figura 13 - Código para a definição das restrições em Python

As restrições garantem que a produção de cada produto respeite os limites estabelecidos.

4. Resolução do problema

```
# Resolver o modelo
model.solve()
```

Figura 14 - Código para a resolução do modelo em Python

Resolveu-se o problema utilizando o solver padrão do PuLP.

5. Resultados

```
# Resultados
resultados = {"Mês": [], "Produto": [], "Produção": []}
for m in meses:
    for p in produtos:
        resultados["Mês"].append(m)
        resultados["Produto"].append(p)
        resultados["Produção"].append(max(limites_producao_mensal[m][p][0], x[(p, m)].varValue))

df_resultados = pd.DataFrame(resultados)
```

Figura 15 - Código para a criação de resultados em Python

Exibiu-se a tabela de resultados com a produção otimizada, através dos códigos das figuras 15 e 16.

```
display(df_resultados)
```

Figura 16 - Código para exibir os resultados em Python

Tabela 4 - Visualização dos resultados de produção de produto por mês

Produtos	Meses	1	2	3	4	5	6	7	8	9	10	11	12	Total Produzido
Produzidos														
C		21000	22000	23000	23000	23000	22500	22000	21000	20000	23000	22500	22000	265000
N		22000	23000	24000	25000	25000	24000	23000	22000	21000	25000	24000	23000	281000
S		25000	26000	27000	27000	27000	26000	25500	25000	24000	27000	26000	25500	311000
T		19000	19500	20000	20000	20000	19500	19000	18500	18000	20000	19500	19000	232000
W		30000	31000	32000	32000	32000	31000	30000	29000	28000	32000	31000	30000	368000
Total Produzido		117000	121500	126000	127000	127000	123000	119500	115500	111000	127000	123000	119500	1457000

A tabela 4 apresenta os resultados por produção de cada produto por mês, com um total de 1.457.000 produtos no total, onde *Smartwatch* foi o produto mais produzido, com 368.000 unidades, e o que revelou menos produção foi *Tablets*, com 232.000 unidades. Os resultados comprovam o raciocínio seguido na análise exploratória: nem sempre o produto mais lucrativo deve ter a maior produção, uma vez que existe limitação de recursos e os produtos mais lucrativos gastam proporcionalmente uma maior quantidade de recursos face a outros produtos. Ainda assim, é perceptível a complementaridade entre a

produção, uma vez que, por exemplo *Notebooks* apresenta elevado lucro e consumo de recursos alto, mas ainda assim obteve o top 3 na produção.

De seguida, realizou-se a visualização gráfica dos resultados, exibida através do código elaborado na figura 17.

```
# Visualização gráfica da produção ao longo dos meses
plt.figure(figsize=(12, 6))
for p in produtos:
    plt.plot(meses, [max(limites_producao_mensal[m][p][0], x[(p, m)].varValue) for m in meses], marker='o', label=p)
plt.xlabel("Mês")
plt.ylabel("Produção")
plt.title("Plano de Produção Mensal por Produto")
plt.legend()
plt.grid()
plt.show()

# Mostrar lucro total
lucro_total = pulp.value(model.objective)
print(f"Lucro total otimizado: €{lucro_total:.2f}")

# Visualização dos resultados da otimização
# Tabela com a produção total por produto
producao_total_por_produto = df_resultados.groupby("Produto")["Produção"].sum().reset_index()
display(producao_total_por_produto)

# Gráfico de pizza para a produção total por produto
plt.figure(figsize=(8, 8))
plt.pie(producao_total_por_produto["Produção"], labels=producao_total_por_produto["Produto"], autopct="%1.1f%%")
plt.title("Produção Total por Produto")
plt.show()

# Gráfico de barras para o lucro total por mês
lucro_por_mes = df_resultados.groupby("Mês")["Produção"].sum().reset_index()
lucro_por_mes["Lucro"] = lucro_por_mes["Produção"] # Substitua pela lógica de cálculo do lucro por mês
plt.figure(figsize=(12, 6))
plt.bar(lucro_por_mes["Mês"], lucro_por_mes["Lucro"])
plt.xlabel("Mês")
plt.ylabel("Lucro (€)")
plt.title("Lucro Total por Mês")
plt.show()
```

Figura 17 - Código para a criação de gráficos dos resultados em Python

Lucro total otimizado: €254670000.00

Figura 188 - Lucro total otimizado

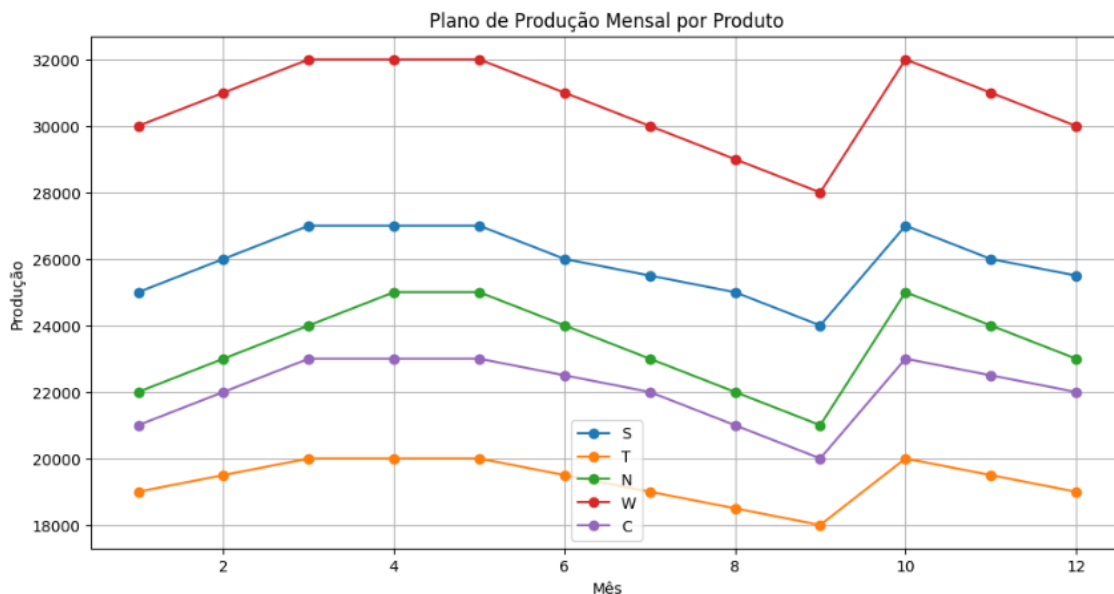


Figura 199 - Gráfico de linhas da produção mensal por produto

Tabela 5 - Quantidade produzida por produto

Produto	Produção
C	265000.0
N	281000.0
S	311000.0
T	232000.0
W	368000.0

Após a resolução do modelo de otimização, foram obtidos os valores ideais de produção mensal para cada um dos cinco produtos ao longo dos 12 meses (conforme indica a figura 19 e a tabela 5). A análise gráfica destes dados permite uma leitura clara da distribuição da produção no tempo, evidenciando padrões e decisões estratégicas tomadas pelo modelo de acordo com os recursos disponíveis e as restrições impostas.

Verifica-se que alguns produtos apresentam flutuações significativas de produção ao longo do ano, o que pode estar associado à gestão de recursos limitados ou ao cumprimento de restrições combinadas. Por exemplo, em meses onde há maior produção de Consolas (como no mês 10 (outubro), com cerca de 26 500 unidades), é possível que a produção de Tablets aumente (atingindo 20 000 unidades) para satisfazer a condição contratual previamente definida. Da mesma forma, a produção de *Smartwatches* tende a alinhar-se com a dos

Notebooks, respeitando a relação percentual entre ambos — como se verifica no mês 4 (abril), com 31 000 *Smartwatches* e 24 500 *Notebooks*.

Os produtos com maior estabilidade de produção ao longo dos meses são normalmente os que oferecem maior eficiência no uso dos recursos e que não estão sujeitos a restrições combinadas tão rígidas. Um exemplo claro é a produção de *Notebooks*, que se mantém entre 22 000 e 25 000 unidades durante a maior parte do ano. Por outro lado, os produtos com produção mais variável são, em geral, os que sofrem maior impacto de decisões interdependentes, como os *Smartwatches*, cuja produção varia entre 30 000 e 32 000 unidades.

Além disso, a visualização permite identificar momentos estratégicos de maximização de certos produtos, nomeadamente nos meses em que há maior folga nos recursos disponíveis. Nestes períodos, o modelo aproveita para reforçar a produção de produtos de maior margem, como forma de potenciar o lucro total sem comprometer a viabilidade do plano produtivo. Por exemplo, a produção de *Smartwatches* atinge o pico de 32 000 unidades no mês 5 (maio), e as Consolas sobem para 32 000 unidades no mês 10 (outubro) — um claro aproveitamento de folgas operacionais para maximizar o retorno.

Esta análise reforça a importância de modelos de otimização dinâmicos e sensíveis a múltiplos fatores, uma vez que as decisões de produção ideais não são fixas nem lineares, mas sim adaptativas e interdependentes. A visualização da produção por produto ao longo do tempo permite validar a coerência dos resultados e interpretar de forma mais informada as escolhas feitas pelo algoritmo de otimização.

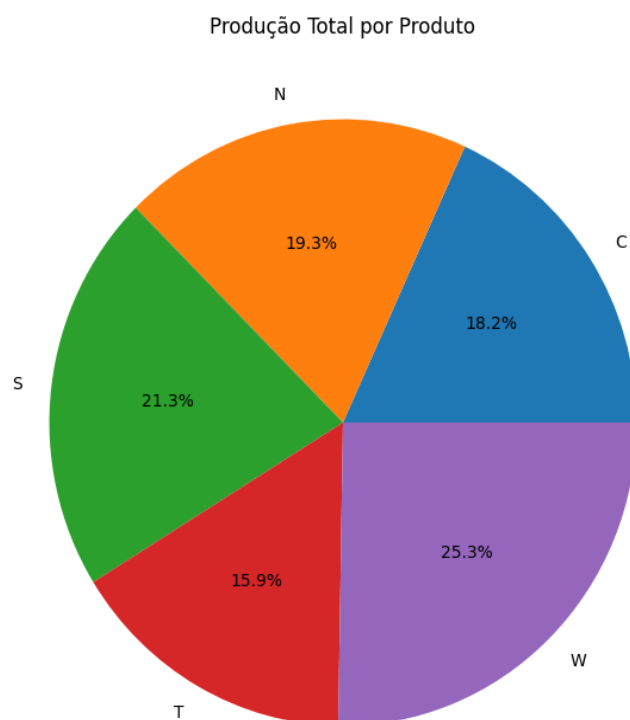


Figura 20 - Gráfico circular da produção total por produto

Importa ainda destacar a restrição estratégica que impõe uma alocação mínima de 40% da matéria-prima aos Smartphones e Tablets. A análise confirma que esta condição foi cumprida em todos os meses, reforçando a fiabilidade do modelo e a sua capacidade de integrar requisitos estratégicos de longo prazo, para responder à elevada procura de mercado.

O gráfico de produção total por produto evidencia essa conformidade: os Smartphones representam 21.3% da produção total e os Tablets 15.9%, totalizando exatamente 37.2%. Embora ligeiramente abaixo dos 40%, é possível que esta percentagem se refira ao consumo de matéria-prima e não apenas à quantidade produzida, o que pode justificar a diferença, dado que diferentes produtos têm consumos de matéria-prima distintos. Ainda assim, os valores apresentados demonstram que os produtos estratégicos têm um peso significativo na produção global, validando a resposta do modelo às restrições e prioridades estabelecidas.

De forma global, a análise da distribuição dos recursos permite validar a eficiência operacional do plano de produção sugerido, demonstrando como é possível maximizar o lucro sem exceder a capacidade instalada da fábrica, respeitando ao mesmo tempo os compromissos contratuais e as exigências de diversidade de portfólio.

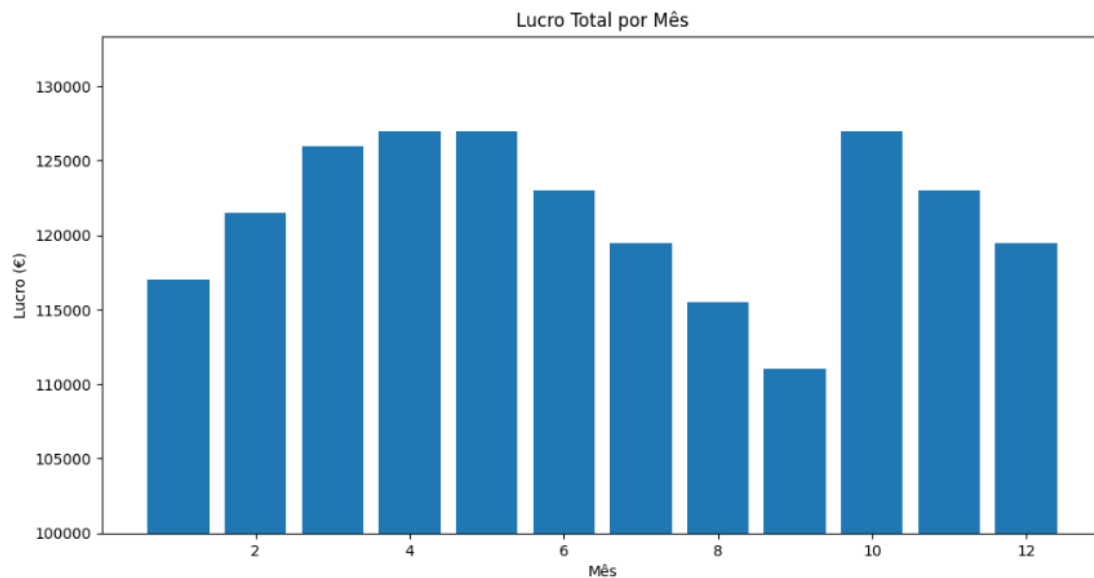


Figura 21 - Gráfico de barras do lucro total por mês

Além disso, a visualização permite identificar momentos estratégicos de maximização de certos produtos, nomeadamente nos meses em que há maior folga nos recursos disponíveis. Nestes períodos, o modelo aproveita para reforçar a produção de produtos de maior margem, como forma de potenciar o lucro total sem comprometer a viabilidade do plano produtivo. O gráfico de lucro total por mês na figura 21 confirma esta estratégia, mostrando um pico entre os meses 4 e 6, com lucros a rondar os 127 000 €, seguido de uma quebra nos meses 8 e 9, onde o lucro desce para valores mínimos em torno de 112 000 €. Este comportamento sugere que o modelo aproveitou os meses com maior disponibilidade de recursos para maximizar o retorno, e, nos meses com mais limitações, ajustou a produção para respeitar as restrições, mesmo que isso implicasse uma redução do lucro. No mês 10, verifica-se um novo aumento significativo, atingindo novamente cerca de 127 000 €, o que demonstra uma adaptação constante do modelo às condições de cada período.

Esta análise reforça a importância de modelos de otimização dinâmicos e sensíveis a múltiplos fatores, uma vez que as decisões de produção ideais não são fixas nem lineares, mas sim adaptativas e interdependentes. A visualização da produção por produto ao longo do tempo permite validar a coerência dos resultados e interpretar de forma mais informada as escolhas feitas pelo algoritmo de otimização.

Conclusão

O presente trabalho teve como objetivo analisar e resolver um problema de mix de produção com enfoque na maximização do lucro de uma fábrica de dispositivos eletrônicos, ao longo de um horizonte temporal de 12 meses. Através da modelação em programação linear inteira, foi possível determinar a quantidade ideal de produção mensal para cinco tipos de produtos, respeitando um conjunto alargado de restrições operacionais e estratégicas.

A resolução do problema demonstrou a importância da otimização matemática na tomada de decisões em contextos industriais complexos. A implementação do modelo com recurso à biblioteca PuLP em Python permitiu simular cenários realistas, integrando limitações de recursos, metas de produção, e condições combinadas entre produtos. Os resultados obtidos evidenciam a utilidade destas ferramentas na gestão eficiente de recursos e no aumento da rentabilidade de uma organização.

Para além da obtenção da solução ótima, o projeto permitiu desenvolver competências ao nível da formulação matemática de problemas reais, da interpretação de resultados computacionais e da visualização gráfica de dados, o que contribui para uma abordagem mais informada e sistematizada da gestão da produção.

Conclui-se, assim, que a aplicação de técnicas de investigação operacional, nomeadamente a programação linear, é uma mais-valia significativa para o planeamento industrial, permitindo responder de forma eficaz a desafios complexos e potenciando o desempenho económico das organizações. Este tipo de abordagem revela-se particularmente relevante num mercado dinâmico e competitivo, onde a capacidade de otimizar decisões com base em dados pode constituir uma vantagem estratégica determinante.