

// laboratório codex 05 · microeconomia 4.0

python + ia + app

>_ Laboratório Codex 05

A micro inteligente, com IA

Banco de dados, previsão de preços, otimização numérica, painel interativo, alertas e relatório executivo: a microeconomia que ajuda a decidir.

// apostila do aluno V · o projeto em etapas

01 Banco de dados

03 Otimização numérica

05 Alertas

07 Monopólio e Cournot

02 Previsão de preços

04 Painel do consumidor

06 Relatório executivo

08 Apresentação



Prof. Caio Haffner

Mestre em Desenvolvimento Econômico · Unicamp

1 Como usar esta apostila

Esta é a **Apostila V de Microeconomia**, a etapa avançada da trilha: ela continua a **Apostila IV** (o Simulador de Microeconomia no ar (Apostila IV)). Agora você vai dar **inteligência** ao Simulador de Microeconomia: um banco de dados com os preços, a previsão do preço, a otimização numérica do consumidor e da empresa, um painel interativo, alertas automáticos e um relatório executivo.

1. Guardar

Banco de dados

2. Prever

Preço médio de um produto

3. Otimizar

Escolha do consumidor

4. Vigiar

Alertas

5. Decidir

Relatório executivo



O PROJETO: A MICRO INTELIGENTE

Nas Apostilas III e IV, o simulador resolve os modelos por fórmulas. Agora ele passa a otimizar como o computador faz: acha a melhor escolha por busca numérica, prevê preços, avisa quando algo sai do padrão e escreve o relatório do mês.

Ao final, você deverá conseguir

Habilidade	Como saber que consegui	Ok
Usar um banco de dados	O histórico fica num banco SQLite e o app consulta com SQL.	☐
Prever	Comparo média móvel e tendência e meço o erro.	☐
Otimizar numericamente	Acho a cesta ótima e o lucro máximo com scipy.	☐
Mostrar num painel	Os resultados aparecem num painel interativo.	☐
Criar alertas	O app avisa quando um número sai do padrão.	☐
Relatar	O app gera o relatório executivo do mês.	☐

1

REGRA DE OURO

Você é responsável pelo código que entrega, mesmo quando a IA o escreveu. **Não entregue código que você não consegue explicar.** Aqui vale mais uma: **previsão, simulação e otimização são apoio à decisão.** Quem decide continua sendo uma pessoa, que entende os números.



LEIA TUDO, NA ORDEM

Leia esta apostila **desde a primeira palavra** e faça tudo o que ela manda, na ordem. Cada etapa usa o que a anterior construiu.

A Apostila V começa onde a IV termina: **o Simulador de Microeconomia no ar, no GitHub, com testes e dados reais**. Se ainda não está assim, termine a IV antes.

Instale as bibliotecas novas (uma vez só)

TERMINAL

```
python -m pip install numpy scipy plotly fpdf2
```

Biblioteca	Para que serve
sqlite3	Banco de dados num arquivo (já vem com o Python).
numpy e scipy	Contas, simulação e otimização (a melhor decisão dentro dos limites).
plotly	Gráficos interativos dentro do app: zoom, filtro e valor ao passar o mouse.
fpdf2	Gerar o relatório executivo em PDF.

Ao longo das etapas, a pasta do projeto vai ganhar:

TERMINAL

```
Laboratorio_Codex_MICR03/  
├─ dados/historico.db ← novo: banco de dados  
├─ calculos.py  
├─ previsao.py        ← novo  
├─ otimizacao.py      ← novo  
├─ painel.py          ← novo  
├─ alertas.py         ← novo  
├─ app.py  
└─ tests/
```



LEMBRETES DAS APOSTILAS ANTERIORES

Peça sempre um **plano** antes do código · teste com contas feitas por você · peça o **diagnóstico antes da correção** · nada vai para a internet sem passar nos testes · nunca coloque senhas no código.

3

Etapa 1 • Guardar: o banco de dados

Planilhas soltas não aguentam anos de histórico. Um **banco de dados** guarda tudo e responde perguntas em segundos.

PROMPT 1

Crie dados/historico.db (SQLite, que já vem com o Python) com as tabelas precos (mês, produto, preço), consumidores (renda, cesta) e mercado (preço, quantidade demandada, ofertada),

importando os dados das Apostilas III e IV. Crie banco.py com funções que usem SQL para:

1. o preço médio de cada produto por trimestre;
2. a variação (%) do preço mês a mês;
3. os meses com excesso de demanda (demandada maior que ofertada).

Antes de criar, explique em 5 linhas o que são tabela, linha, coluna e consulta SQL.



CHECKPOINT

Os preços do banco são os mesmos da aba "Preços reais" da Apostila IV.



LEIA O SQL

Peça ao Codex: "explique esta consulta SQL linha por linha". SELECT escolhe as colunas, WHERE filtra as linhas, GROUP BY agrupa e ORDER BY ordena.

4 Etapa 2 · Prever: o que vem por aí?

Para decidir o próximo mês, é preciso uma **previsão**. Duas formas simples e honestas: média móvel e tendência.



média móvel de 3 períodos = média dos 3 últimos ·
tendência: valor = a + b × período (regressão)

Erro absoluto médio (EAM) = média de $|\text{real} - \text{previsto}|$ nos períodos em que dá para testar



PREÇO MÉDIO DE UM PRODUTO (R\$, 12 MESES)

10,00 · 10,20 · 10,10 · 10,40 · 10,60 · 10,50 · 10,80 · 11,00 · 10,90 · 11,20 · 11,40 · 11,30

PROMPT 2

Crie previsao.py com:

1. media_movel(serie, n=3) e a previsão do período 13;
2. tendencia(serie): a e b da reta por mínimos quadrados e a previsão dos períodos 13 e 14;
3. o erro absoluto médio da média móvel nos períodos 4 a 12.

No app, mostre um gráfico com o histórico, as duas previsões e uma frase dizendo qual erra menos.

Resultado	Esperado	Calculado
Média móvel (período 13)	11,30	
Tendência: a e b	$a \approx 9,86$ · $b \approx 0,129$	
Tendência (períodos 13 e 14)	11,54 e 11,67	
Erro absoluto médio da média móvel	0,27	



PARA PENSAR

O preço sobe $\approx$ R\$ 0,13 por mês. Isso é aumento real ou só inflação? Como a aba "Preços reais" da Apostila IV ajuda a responder?

5

Etapa 3 · Otimizar: a escolha do consumidor

Na Apostila III, a cesta ótima saiu de uma fórmula. Agora o computador **procura** a melhor cesta, como faria com qualquer função.



O CONSUMIDOR

Utilidade $U = x^{0,4} \cdot y^{0,6}$ · renda R\$ 1.000 · preço de x R\$ 10 · preço de y R\$ 20.

PROMPT 3

Crie `otimizacao.py` com `scipy.optimize.minimize`:

1. a cesta (x, y) que dá a maior utilidade gastando no máximo a renda;
2. a conferência pela fórmula de Cobb-Douglas da Apostila III;
3. o que acontece com a cesta quando o preço de x vai de R\$ 5 a R\$ 20 (a curva de demanda de x).

Explique antes, em 5 linhas, por que minimizamos "menos a utilidade".

Resultado	Esperado	Calculado
Cesta ótima (x, y)	(40, 30)	
Utilidade máxima	≈ 33,66	
Quantidade de x com preço R\$ 20	20	



PARA PENSAR

Com Cobb-Douglas, o consumidor gasta sempre 40% da renda em x. Como isso aparece na curva de demanda que o app desenhou?

Etapa 4 · Mostrar: o painel do consumidor

Ver a curva de indiferença tocando a reta do orçamento ensina mais que a fórmula.

PROMPT 4

Acrescente ao app a aba "Consumidor" com plotly: a reta do orçamento, três curvas de indiferença e a cesta ótima marcada; controles deslizantes para a renda e os preços. Ao lado, a curva de demanda de x da Etapa 3. Ao passar o mouse, mostre a cesta e a utilidade.



CHECKPOINT

Com renda R\$ 1.000, P_x R\$ 10 e P_y R\$ 20 o ponto ótimo fica em (40, 30) e se move quando um controle muda.

7

Etapa 5 · Vigiar: alertas automáticos

Ninguém olha todos os números todos os dias. Os **alertas** mostram só o que pede atenção.

PROMPT 5

Crie `alertas.py` com regras que o usuário pode ajustar na barra lateral:

1. preço com alta de mais de 2% no mês;
2. aumento de preço num produto de demanda elástica ($|elasticidade|$ maior que 1);
3. cesta escolhida acima da renda;
4. preço previsto acima de R\$ 11,50.

No topo do app, mostre os alertas em vermelho (crítico) e amarelo (atenção), do mais grave para o menos grave.

Alerta	Quantos apareceram	Faz sentido?
Alta acima de 2% no mês		
Aumento com demanda elástica		
Cesta acima da renda		
Previsão acima de R\$ 11,50		



ALERTA DEMAIS É ALERTA NENHUM

Se aparecerem 30 alertas, ninguém olha mais. Ajuste as regras para mostrar só o que exige decisão.

Etapa 6 · Decidir: o relatório executivo

Quem decide não quer tabelas: quer saber o que aconteceu, o que vem aí e o que fazer.

PROMPT 6

Instale fpdf2 (python -m pip install fpdf2) e acrescente o botão "Relatório executivo (PDF)",
de uma página: os preços do mês e a variação real, a previsão dos próximos 2 meses,
a cesta ótima com os preços atuais, os 3 alertas mais graves e 3 recomendações
escritas pelo app,
cada uma com o número que a justifica. Atualize os testes e publique a nova versão
(como na Apostila IV).



CHECKPOINT

O PDF tem uma página, os números batem com o app e a nova versão está no ar pelo link da Apostila IV.

9

Desafio extra · O monopolista numérico

A mesma ferramenta resolve o problema da empresa: qual quantidade dá o maior lucro?

PROMPT 7

desafio

Use `scipy.optimize.minimize_scalar` para achar a quantidade que maximiza o lucro do monopolista com demanda $P = 100 - 2Q$ e custo marginal 20. Depois, resolva o duopólio de Cournot por iteração: cada empresa escolhe a melhor quantidade dada a da outra, até parar de mudar. Compare preço, quantidade e lucro dos dois casos. Explique o resultado em 3 linhas antes de mostrar o código.



CHECKPOINT

Monopólio: $Q = 20$, $P = 60$, lucro 800 · Cournot com 2 empresas: $Q \approx 13,33$ cada, $P \approx 46,67$.

10

Use o Codex como revisor do projeto

REVISAR A ORGANIZAÇÃO

Revise o meu projeto como um programador experiente revisaria o de um iniciante. Não faça alterações. Aponte: funções muito longas, código repetido, nomes pouco claros e cálculos misturados com a tela. Separe o que é obrigatório do que é opcional.

REVISAR O CONTEÚDO

Revise os cálculos do meu projeto como um professor de Microeconomia. Não faça alterações. Confira o SQL, a previsão (e o erro medido), a restrição do orçamento na otimização e as regras dos alertas e liste o que pode estar errado.



GRÁFICO BONITO NÃO É GRÁFICO CERTO

Confira pelo menos um número de cada gráfico com a calculadora ou com a Apostila II. A IA erra eixos, unidades e somas com frequência.

Entrega: apresente o seu projeto

Um projeto só existe quando alguém o vê funcionando. Prepare uma **apresentação de 3 minutos** e um registro para o seu portfólio.

Roteiro de 3 minutos

Tempo	O que mostrar
30 s	O problema que o app resolve e para quem.
1 min 30 s	O app funcionando: mexa nos controles e mostre a cesta ótima se movendo quando os preços mudam.
30 s	Um erro que a IA cometeu e como você descobriu.
30 s	O que você faria a seguir, se tivesse mais tempo.

O que entregar

- ☐ A pasta do projeto, com todos os arquivos .py e os dados
- ☐ O relatório em Excel gerado pelo app
- ☐ Três prints do app (telas principais)
- ☐ Um arquivo LEIAME.md explicando o que o app faz e como rodar (peça um rascunho ao Codex e revise)
- ☐ O diário de prompts: pelo menos 5 prompts importantes e o que aconteceu



PARA O SEU PORTFÓLIO

Um post no LinkedIn com um vídeo curto do app funcionando ("Construí em Python, com ajuda de IA, um app de microeconomia com otimização numérica") mostra a recrutadores algo que poucos estudantes têm: **uma ferramenta real feita por você.**

Reflexão individual

1. Qual etapa foi a mais difícil, e como você a resolveu?

2. Em que momento a IA errou, e como você percebeu?

3. O que você aprendeu de Microeconomia usando previsão e otimização que não tinha aprendido nas apostilas anteriores?

12 Como você será avaliado

Critério	0	1	2	3
O app funciona	Não abre	Abre com erros	Funciona	Funciona e trata entradas inválidas
Conteúdo	Cálculos errados	Alguns certos	Quase todos certos	Todos certos e conferidos
Gráficos	Nenhum	Sem títulos ou eixos	Corretos	Corretos e fáceis de ler
Relatório	Não gera	Gera incompleto	Gera completo	Completo e organizado em abas
Uso do Codex	Não usou	Só copiou	Usou com compreensão	Usou de forma crítica
Apresentação	Não apresentou	Confusa	Clara	Clara, no tempo e com o erro da IA

Total máximo: 18 pontos, convertidos para a escala da disciplina.

13 Consulta rápida

Comandos

TERMINAL

```
python -m pip install numpy scipy
plotly fpdf2
python -m pytest                # roda
os testes
python -m streamlit run app.py  # abre
o app
git add . ; git commit -m "..."; git
push # publica
```

Streamlit

<code>st.title · st.header</code>	Títulos
<code>st.number_input · st.slider</code>	Entradas de número
<code>st.selectbox · st.file_uploader</code>	Escolha e upload
<code>st.metric · st.dataframe</code>	Indicador e tabela
<code>st.pyplot(fig)</code>	Mostrar um gráfico
<code>st.tabs · st.sidebar</code>	Abas e barra lateral
<code>st.download_button</code>	Botão para baixar o Excel

Glossário

Biblioteca

Código pronto que você instala para ganhar recursos (gráficos, telas).

pip

O instalador de bibliotecas do Python.

DataFrame

Tabela do pandas, com linhas e colunas.

App web

Programa que roda no navegador.

SQLite

Banco de dados num arquivo só, sem servidor.

SQL

Linguagem para consultar tabelas (SELECT, WHERE, GROUP BY).

Média móvel

Média dos últimos períodos, usada para prever o próximo.

Erro absoluto médio

Quanto a previsão erra, em média, para mais ou para menos.

Otimização numérica

O computador procura a melhor escolha por tentativas inteligentes, sem fórmula.

Restrição orçamentária

Gasto total que não pode passar da renda.

1

LEMBRE-SE

A IA é uma ferramenta. Conhecimento, julgamento, validação e responsabilidade continuam sendo de quem programa. Nada nesta apostila é recomendação profissional: é aprendizado.