

Do Imóvel ao Estoque: Aplicando o CLTV com Python na Gestão de Riscos Empresariais



CLTV - Uma Ideia Empréstada

Já imaginou usar uma métrica famosa do mercado imobiliário, o CLTV, para avaliar os riscos financeiros da sua empresa? O CLTV - Combined Loan-to-Value, conhecido por medir a segurança de financiamentos imobiliários, pode ser adaptado para ajudar gestores a analisarem sua capacidade de assumir novos contratos usando o estoque como garantia.

Neste artigo, apresentarei um script em Python para calcular o CLTV empresarial, automatizar análises financeiras e gerar

relatórios detalhados, facilitando decisões estratégicas. Vamos conferir essa abordagem inovadora?

O que é o CLTV ?

O CLTV (Combined Loan-to-Value) é uma métrica versátil, amplamente aplicada em diferentes setores para avaliar riscos, retornos financeiros e planejamento estratégico. Originalmente conhecido por sua aplicação no mercado imobiliário, o CLTV também é utilizado para medir o valor da vida útil do cliente em negócios voltados à recorrência e, mais recentemente, emprestamos esse conceito para analisar o endividamento empresarial com garantia de estoque. Vamos entender como isso funciona.

O CLTV e Suas Aplicações

1. CLTV no Valor da Vida Útil do Cliente

No contexto de retenção de clientes, o CLTV mede a receita potencial que um cliente pode gerar ao longo de sua vida como consumidor ativo. Ele considera fatores como:

- Vida útil média do cliente (por quanto tempo o cliente permanece ativo).
- Frequência média de compras (quantas vezes o cliente compra em determinado período).
- Valor médio de compra (quanto, em média, o cliente gasta por compra).

Fórmula do CLTV no contexto do cliente:

$$CLTV = \frac{\text{Valor Médio do Cliente}}{\text{Vida Útil Média do Cliente}}$$

Essa métrica ajuda empresas a prever receitas futuras e identificar oportunidades para melhorar a retenção e a lucratividade.

2. CLTV no Mercado Imobiliário

No setor imobiliário, o CLTV é utilizado para medir o risco de financiamento, comparando o valor de todos os empréstimos garantidos por um imóvel ao valor de mercado do próprio imóvel.

Fórmula do CLTV no setor imobiliário:

$$CLTV = \frac{\text{Soma dos Empréstimos Garantidos}}{\text{Valor de Mercado do Imóvel}} \times 100$$

Essa aplicação garante que o valor do imóvel seja suficiente para cobrir os empréstimos, ajudando bancos a avaliarem se o mutuário representa um risco aceitável.

3. CLTV Aplicado ao Endividamento Empresarial

Emprestamos o conceito do CLTV imobiliário para analisar a capacidade de uma empresa de honrar compromissos financeiros, usando o estoque como garantia. A lógica aqui é simples: se o estoque for suficiente para cobrir os contratos firmados, o risco financeiro é controlado.

Fórmula adaptada para o setor empresarial:

$$CLTV = \frac{\text{Soma dos Contratos Bancários}}{\text{Valor Atual do Estoque}} \times 100$$

- **Soma dos Contratos Bancários:** Total dos compromissos financeiros assumidos pela empresa.
- **Valor Atual do Estoque:** O valor estimado que o estoque pode gerar caso seja vendido.

Essa métrica funciona como um indicador de risco para gestores avaliarem:

1. Se a empresa pode assumir novos contratos.
2. Qual o impacto do endividamento sobre sua saúde financeira.

Comparação Entre Setores

Para compreender melhor como o CLTV pode ser adaptado para diferentes contextos, veja um quadro comparativo entre o setor imobiliário e o setor empresarial. Ele destaca as principais semelhanças e diferenças, facilitando a visualização da aplicação desse conceito em cada área.

Confira:

Aplicação do CLTV	Setor Imobiliário	Setor Empresarial
Garantia	Valor de mercado do imóvel	Valor atual do estoque
Empréstimos ou Riscos	Soma dos empréstimos garantidos	Soma dos contratos bancários
Objetivo	Medir o risco de financiamento	Avaliar o risco de endividamento empresarial
Indicador de Risco	CLTV > 100% = alto risco	CLTV > 100% = risco de inadimplência
Tomador de Decisão	Bancos	Gestores e analistas financeiros

Fonte: O Autor

Por Que Utilizar o CLTV no Setor Empresarial?

1. **Avaliação de Riscos:** Ajuda a empresa a prever sua capacidade de honrar contratos.
2. **Planejamento Estratégico:** Oferece uma visão clara sobre o impacto de novos contratos.
3. **Negociações Bancárias:** Permite demonstrar objetivamente a saúde financeira da empresa.

Tomada de Decisões Consciente: Evita o excesso de endividamento, garantindo maior segurança financeira.

Adaptação do CLTV com Python

Para demonstrar a viabilidade desse conceito no setor empresarial, desenvolvemos um script em Python. Ele:

1. Automatiza os cálculos do CLTV.
2. Exibe o impacto de novos contratos financeiros.
3. Gera relatórios profissionais, facilitando a análise e a comunicação dos resultados.

A Automação em Python

Para validar a aplicação do CLTV no contexto empresarial, foi desenvolvido um **script em Python** que automatiza:

1. A análise dos contratos existentes.
2. O cálculo do impacto de novos contratos.
3. A geração de relatórios com análises detalhadas.

Por que usar Python?

- Python é uma linguagem versátil e amplamente utilizada em finanças.
- Possui bibliotecas poderosas para cálculos, formatações e geração de relatórios (ex.: fpdf e tabulate).
- Permite criar ferramentas práticas, escaláveis e amigáveis.

Arquivos do Script

O projeto foi organizado em três arquivos principais:

1. **simulacao.py**

- Contém as funções principais para o cálculo do CLTV.
- Calcula o CLTV atual, o CLTV projetado e a margem de segurança.
- Fornece alertas sobre o nível de risco (seguro ou alto risco).

2. **gerar_relatorio.py**

- Gera relatórios profissionais em PDF.
- Inclui os resultados, tabelas e um parecer técnico com recomendações detalhadas.

3. **main.py**

- Centraliza a execução do script.
- Solicita os dados do analista e da empresa.
- Exibe os resultados no terminal e chama a função para gerar o relatório.

Explicação do Funcionamento

1. **Entrada de Dados**

- O analista informa:
 - Valores dos contratos existentes.
 - Valor do estoque atual.
 - Valor do novo contrato.
 - Percentual de capacidade de venda do estoque.

2. **Cálculos Realizados**

- CLTV atual: Considera os contratos existentes.
- CLTV projetado: Adiciona o impacto do novo contrato.
- Capacidade de venda líquida: Projeta quanto do estoque pode ser convertido em receita.
- Margem de segurança: Avalia a diferença entre a capacidade de venda líquida e os compromissos totais.

3. **Geração do Relatório**

- O script organiza as informações em um relatório PDF, incluindo:
 - Local e data da análise.
 - Introdução com o objetivo do relatório.
 - Tabela completa dos resultados.
 - Parecer técnico com recomendações.

Resultados

- **Tabela no Terminal:** Exibe de forma clara os valores informados e os resultados calculados.
- **Relatório em PDF:**
 - Formato profissional, com cabeçalho, tabela e parecer técnico.
 - Adapta-se ao cenário (margem positiva ou negativa).
 - Inclui o nome do analista e suas recomendações.

Conclusão

O CLTV é uma métrica poderosa que vai muito além do mercado imobiliário. Seja medindo o valor da vida útil do cliente, analisando riscos em financiamentos imobiliários ou avaliando endividamento empresarial, sua aplicação oferece insights valiosos para a gestão financeira e estratégica. Ao emprestar esse conceito para o setor empresarial, demonstramos como ele pode ser adaptado para diferentes cenários, fortalecendo a tomada de decisões e promovendo maior segurança econômica.

ANEXO - O Código

a) Simulação.py

```
simulacao.py

def calcular_cltv(contratos, valor_estoque, novo_contrato,
capacidade_venda):
    """
    Calcula o índice CLTV (Atual e Projetado) e analisa a margem de
    segurança.
    """
    if valor_estoque == 0:
        raise ValueError("O valor do estoque não pode ser zero.")

    total_contratos = sum(contratos)
    cltv_atual = (total_contratos / valor_estoque) * 100
    cltv_projetado = ((total_contratos + novo_contrato) / valor_estoque)
* 100
    capacidade_liquida = valor_estoque * (capacidade_venda / 100)
    margem_seguranca = capacidade_liquida - (total_contratos +
novo_contrato)
    alerta = "Alto Risco" if margem_seguranca < 0 else "Seguro"

    return {
        "CLTV Atual (%)": round(cltv_atual, 2),
        "CLTV Projetado (%)": round(cltv_projetado, 2),
        "Capacidade de Venda Líquida (R$)": round(capacidade_liquida, 2),
        "Margem de Segurança (R$)": round(margem_seguranca, 2),
        "Alerta": alerta
    }
```

b) gerar_relatorio.py

```
from fpdf import FPDF
from datetime import datetime

def gerar_relatorio(empresa, analista, resultados, contratos,
valor_estoque, novo_contrato):
    """
    Gera um relatório em PDF com os cálculos e análises do CLTV.
    """
    pdf = FPDF()
    pdf.add_page()
    pdf.set_font("Arial", size=12)

    # Cabeçalho com local e data
```

```

local = "Ponta Grossa - PR"
data = datetime.now().strftime("%d de %B de %Y")
pdf.cell(200, 10, txt=f"{local}, {data}", ln=True, align="R")
pdf.ln(10)

# Endereçamento
pdf.set_font("Arial", "B", size=12)
pdf.cell(200, 10, txt="À", ln=True)
pdf.cell(200, 10, txt=f"Empresa {empresa}", ln=True)
pdf.cell(200, 10, txt="Sr. Administrador", ln=True)
pdf.ln(10)

# Introdução
pdf.set_font("Arial", size=12)
pdf.multi_cell(0, 10, txt=(
    "Conforme sua solicitação, apresento abaixo o meu relatório e  

parecer final sobre "  

    "a análise do índice CLTV (Combined Loan-to-Value) da empresa,  

considerando os "  

    "contratos existentes e o impacto de um novo contrato avaliado."
))
pdf.ln(10)

# Tabela com os resultados
pdf.set_font("Arial", "B", size=12)
pdf.cell(200, 10, txt="Resultados do Simulador CLTV:", ln=True)
pdf.set_font("Arial", size=12)
tabela = [
    ["Descrição", "Valor"],
    ["Valor do Estoque Atual (R$)", f"R$ {valor_estoque:,.2f}"],
    ["Total dos Contratos Existentes (R$)", f"R$
{sum(contratos):,.2f}"],
    ["Novo Contrato Avaliado (R$)", f"R$ {novo_contrato:,.2f}"],
    ["Capacidade de Venda do Estoque (%)", f"{resultados['Capacidade
de Venda Líquida (R$)'] / valor_estoque * 100:.1f}%"],
    ["Capacidade de Venda Líquida (R$)", f"R$ {resultados['Capacidade
de Venda Líquida (R$)']:,.2f}"],
    ["CLTV Atual (%)", f"{resultados['CLTV Atual (%)']}%"],
    ["CLTV Projetado (%)", f"{resultados['CLTV Projetado (%)']}%"],
    ["Margem de Segurança (R$)", f"R$ {resultados['Margem de
Segurança (R$)']:,.2f}"],
    ["Alerta", resultados["Alerta"]],
]

for linha in tabela:
    pdf.cell(100, 10, txt=linha[0], border=1)
    pdf.cell(100, 10, txt=linha[1], border=1, ln=True)

pdf.ln(10)

```



```

# Parecer Técnico
pdf.set_font("Arial", "B", size=12)
pdf.cell(200, 10, txt="Parecer Técnico:", ln=True)
pdf.set_font("Arial", size=12)

if resultados["Margem de Segurança (R$)"] < 0:
    pdf.multi_cell(0, 10, txt=(
        f"A margem de segurança é negativa (-R$
{abs(resultados['Margem de Segurança (R$)']):,.2f}), "
        "indicando que a empresa não possui capacidade financeira
suficiente para honrar "
        "os contratos firmados e o novo contrato avaliado.\n\n"
        "Recomendações:\n"
        f"1. Não firmar o novo contrato de R$ {novo_contrato:,.2f}
neste momento.\n"
        "2. Buscar alternativas para aumentar a liquidez do
estoque.\n"
        "3. Renegociar os contratos existentes ou avaliar fontes de
receita adicionais."
    ))
else:
    pdf.multi_cell(0, 10, txt=(
        f"A margem de segurança é positiva (R$ {resultados['Margem de
Segurança (R$)']:,.2f}), "
        "indicando que a empresa possui capacidade financeira
suficiente para honrar "
        "os contratos firmados e o novo contrato avaliado.\n\n"
        "Recomendações:\n"
        "1. Fimar o novo contrato com monitoramento contínuo da
liquidez.\n"
        "2. Planejar adequadamente o uso do estoque para manter a
segurança financeira."
    ))

# Espaço antes do nome do analista
pdf.ln(20)

# Nome do Analista
pdf.set_font("Arial", "B", size=12)
pdf.cell(200, 10, txt=f"Atenciosamente,", ln=True, align="L")
pdf.cell(200, 10, txt=f"{analista}", ln=True, align="L")

# Rodapé
pdf.set_y(-30)
pdf.set_font("Arial", size=10)
pdf.cell(0, 10, txt="Relatório gerado automaticamente pelo Simulador
CLTV.", ln=True, align="C")

```

```
# Salvar o PDF
pdf.output("relatorio_cltv.pdf")
```

c) main.py

```
from tabulate import tabulate
import locale
from src.simulacao import calcular_cltv
from gerar_relatorio import gerar_relatorio

def main():
    # Configurar moeda para Real Brasileiro
    locale.setlocale(locale.LC_ALL, "pt_BR.UTF-8")

    # Solicitar dados do analista
    empresa = input("Informe o nome da empresa: ")
    analista = input("Informe o nome do analista financeiro: ")

    # Solicitar dados de entrada
    print("\n=== Informe os dados para simulação ===")
    contratos = []
    while True:
        contrato = input("Informe o valor de um contrato existente (ou pressione Enter para terminar): ")
        if not contrato:
            break
        contratos.append(float(contrato))
    valor_estoque = float(input("Informe o valor atual do estoque (R$): "))
    novo_contrato = float(input("Informe o valor do novo contrato a ser avaliado (R$): "))
    capacidade_venda = float(input("Informe o percentual de capacidade de venda do estoque (%): "))

    # Cálculo do CLTV
    resultados = calcular_cltv(contratos, valor_estoque, novo_contrato, capacidade_venda)

    # Preparar os dados da tabela
    tabela = [
        ["Valor do Estoque Atual (R$)", locale.currency(valor_estoque, grouping=True)],
        ["Total dos Contratos Existentes (R$)", locale.currency(sum(contratos), grouping=True)],
        ["Novo Contrato Avaliado (R$)", locale.currency(novo_contrato, grouping=True)],
        ["Capacidade de Venda do Estoque (%)", f"{capacidade_venda}%"],
```

```

        ["Capacidade de Venda Líquida (R$)",
locale.currency(resultados["Capacidade de Venda Líquida (R$)"],
grouping=True)],
        ["OBS", f"{capacidade_venda}% atribuídos ao Valor do Estoque
Atual"],
        ["CLTV Atual (%)", f"{resultados['CLTV Atual (%)']}%"],
        ["CLTV Projetado (%)", f"{resultados['CLTV Projetado (%)']}%"],
        ["Margem de Segurança (R$)", locale.currency(resultados["Margem
de Segurança (R$)"], grouping=True)],
        ["Alerta", resultados["Alerta"]],
    ]

    # Exibir a tabela no terminal
    print("\nResultados do Simulador CLTV:\n")
    print(tabulate(tabela, headers=["Descrição", "Valor"],
tablefmt="grid"))

    # Gerar relatório PDF
    gerar_relatorio(empresa, analista, resultados, contratos,
valor_estoque, novo_contrato)
    print("\nRelatório gerado: relatorio_cltv.pdf")

if __name__ == "__main__":
    main()

```

Siga-me no LinkedIn: www.linkedin.com/comm/mynetwork/discovery-see-all?usecase=PEOPLE_FOLLOWS&followMember=izairton-oliveira-de-vasconcelos-a1916351

Minha Newsletter, o link para assinar:

<https://www.linkedin.com/build-relation/newsletter-follow?entityUrn=7287106727202742273>

<https://www.linkedin.com/pulse/criando-um-readmemd-padronizado-e-simplificado-em-seus-vasconcelos-yyi2f>

https://github.com/IOVASCON/simulador_cltv.git