

FIAP MBA

RECOMMENDATION SYSTEMS

PROF. ME. JOSÉ LUIZ PAGNOSSIM

JOSÉ LUIZ PAGNOSSIM

PROFESSOR



 profjose.pagnossim@fiap.com.br

linkedin.com/in/jose-luiz-maturana-pagnossim-3454898
(11) 980587052

Formação Acadêmica

- Graduação: Processamento de Dados - Unicamp
- Especialização: Engenharia de Software – Unicamp
- Mestrado: Sistemas de Informação - USP

Experiência Acadêmica

- FIAP - Professor MBA - Sistemas de Recomendação (2021-atual)
- UniDrummond – Professor Graduação (2003-atual)

Vivência na área de TI

- Analista/Programador desde 1996
- Líder técnico de projetos desde 2005
- Gerente de desenvolvimento de aplicações desde 2020

Experiência Profissional

- Accenture do Brasil (2008-atual)
- Brasilprev
- Itautec
- IBM

Pessoal

- Casado, 45 anos, pai da Maria Eduarda e do Pedro
- Apaixonado pela família, churrasco e confraternizações
- Adora esportes, especialmente futebol (palmeirense).

AGENDA DO CURSO - 3 AULAS DE 4H CADA, CARGA HORÁRIA 12H

1 Conceitos e aplicações

- Conceitos gerais, tipos, técnicas e abordagens
- Impactos, principais SRs e suas características
- Aspectos a considerar, desafios e problemas
- Recomendação na área financeira + Case
- Material de Apoio: Case “Abordagem híbrida para SR de notícias”

2 Desenvolvendo um SR

- Cálculo de distância e similaridade
- A técnica de filtro colaborativo
- Algoritmo de recomendação
- Material de Apoio - Case “SR em uma Fintech” – filtro colaborativo, popularidade e ranqueamento.

► Ferramentas: Python (IDLE), SQL-Server, Visual Studio Community (ASP.NET Web Application e C#)

3 Validando um SR

- Apresentação dos trabalhos
- Discussão a respeito das apresentações
- Avaliação de sistemas de recomendação
- Análise de resultados do Case de Notícias
- Evoluções e Tendências nos SRs

Critérios para avaliação na disciplina

- Peso 100% - Projeto de SR (Grupo): Proposta, arquitetura, protótipo, prova de conceito (POC) de um SR, distribuídos em duas entregas:
 - Apresentação da prévia na Aula 3
 - Entrega final no portal em D+7 em relação à última aula

Capítulo 1

Sistemas de Recomendação (SRs), aplicações e
mercado

Capítulo 1

1.1. Definição de Sistemas de Recomendação

DEFINIÇÃO/CONCEITO

Sistemas de Recomendação são softwares nos quais implementa-se uma ou mais técnicas que são capazes de sugerir itens para os usuários (RICCI et al., 2011).

[Francesco Ricci - Recommender Systems Handbook – Springer]

DEFINIÇÃO - EXEMPLO

Caso George Floyd: tropas ocupam ruas dos EUA após protestos violentos contra o racismo



pressionou o pescoço da vítima com o joelho durante uma abordagem na última segunda-feira.

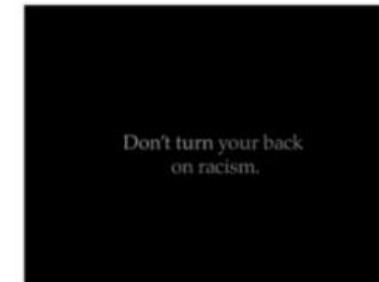
RELACIONADAS



De Obama a Kim Kardashian, personalidades protestam por George Floyd



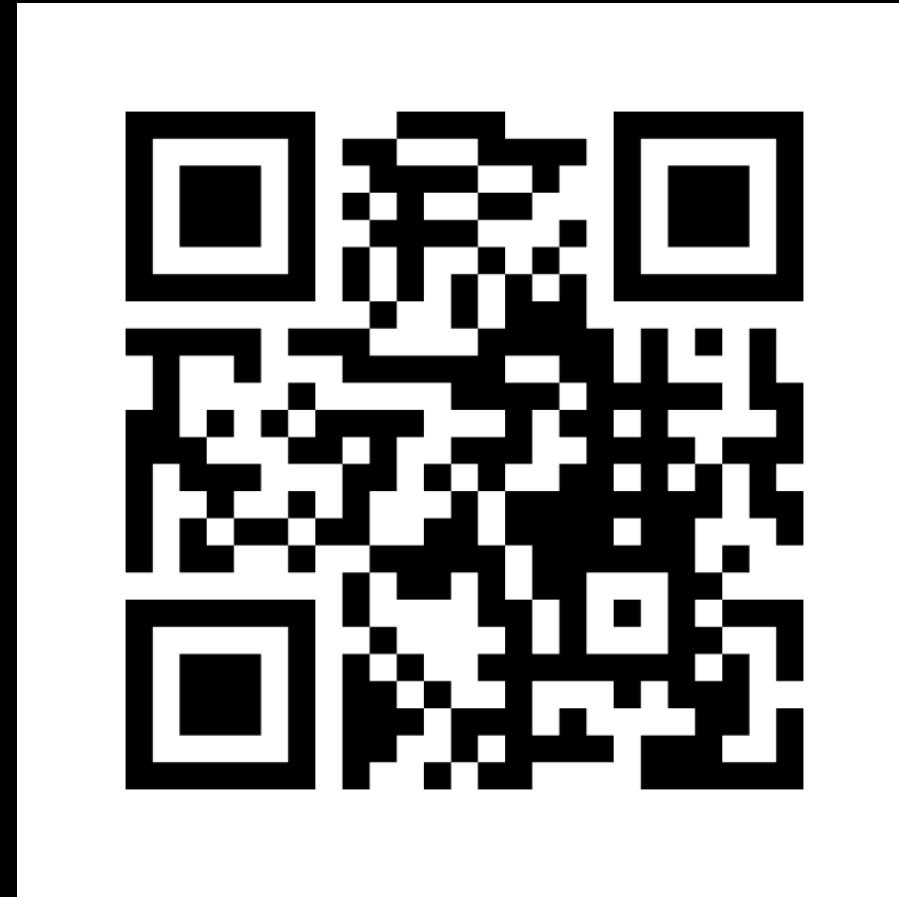
A longa história de segregação e conflito racial em Minneapolis, onde George Floyd foi morto pela polícia



Nike lança manifesto contra o racismo e ganha apoio da Adidas



Indique suas plataformas favoritas

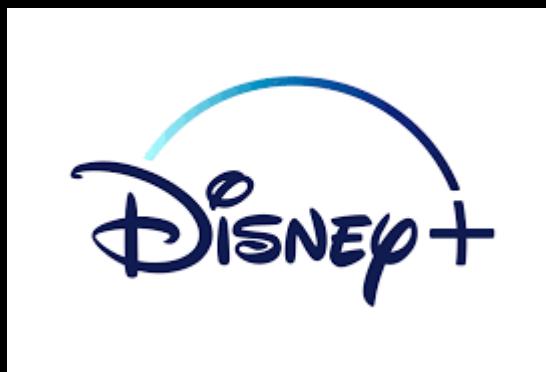


<https://www.menti.com/wxujd139v3>

Capítulo 1

1.2. Aplicações e mercado

FILMES, SÉRIES e DOCUMENTÁRIOS



Características dos Principais SRs

NETFLIX

- Similaridade de Características
- Similaridade entre usuários
- Popularidade
- Features: Surpreenda-me e Relevância.



IMAGEM 3 - NETFLIX

Características dos Principais SRs

NETFLIX

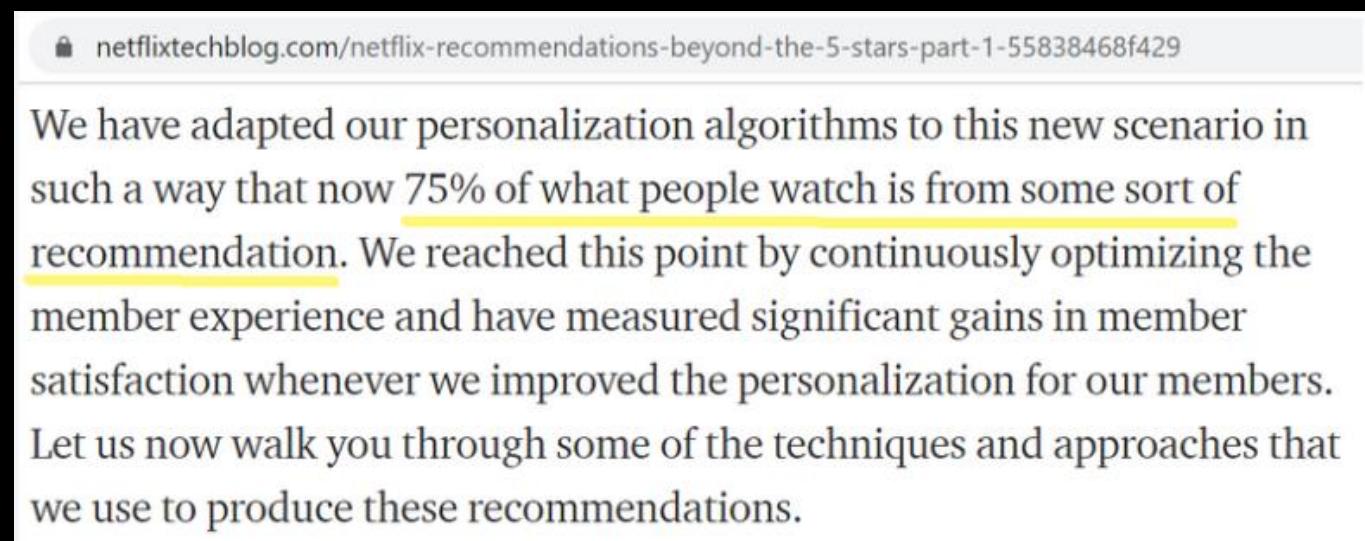


IMAGEM 4 – NETFLIX RECOMENDADOR

Referência: <https://netflixtechblog.com/netflix-recommendations-beyond-the-5-stars-part-1-55838468f429>

COMÉRCIO, PRODUTOS, MERCADO, ALIMENTAÇÃO/ENTREGA



americanas

magalu



阿里巴巴 **Alibaba.com™**
AliExpress



Características dos Principais SRs

MAGALU

- Similaridade de Características
- Similaridade entre usuários
- Popularidade
- Tempo no produto
- Navegação/clicks/favoritos

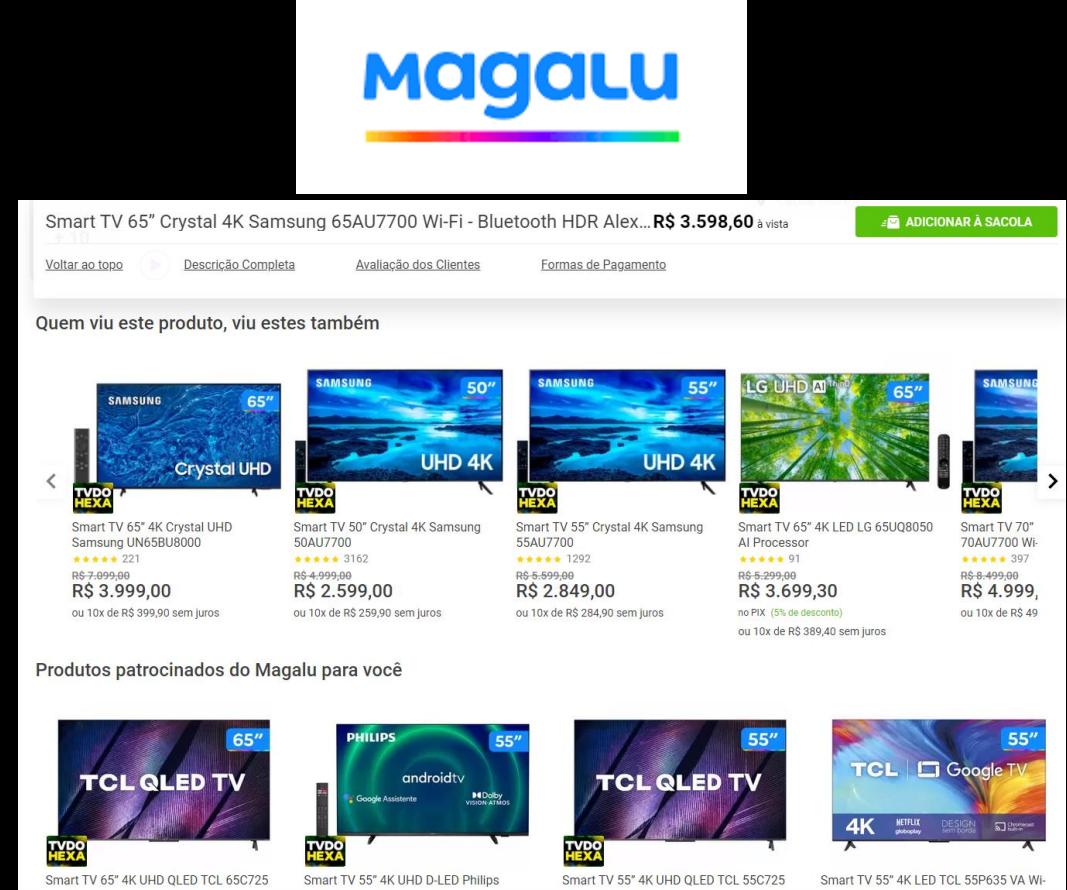


IMAGEM 6 - MAGALU

Características dos Principais SRs

AMAZON

- Similaridade de Características
- Navegação/Clicks
- Compra
- 35% das vendas originadas por recomendação
- 29% aumento em vendas pelo SR (2016)



Entre tantas plataformas de e-commerce, utilizando ferramentas similares, parece-nos que a Amazon é a plataforma de ponta que melhor o faz. Basta mencionar que **35% da receita da Amazon.com** é gerada por seu mecanismo de **recomendação** com uma solução de IA – Inteligência Artificial altamente adaptável.

escoladeecommerce.com/artigos/sucesso-do-sistema-de-recomendacao-da-amazon/

Chama a atenção o fato deles creditarem a seus sistemas de reconhecer a preferência do cliente o **aumento de 29% no total de vendas**, adicionando 135.99 bilhões de reais na venda anual em 2016.

IMAGEM 7 - AMAZON

MÚSICA, RÁDIO, ÁUDIO, VÍDEO



Características dos Principais SRs

YOUTUBE

- Watch time
- Popularidade
- Frequência de postagem do youtuber



IMAGEM 9 - YOUTUBE

Características dos Principais SRs

SPOTIFY

- Similaridade de Características
- Similaridade entre usuários
- Audio time
- Privilegia artistas que publicam a cada 4 semanas
- Feature: Enriquecimento de playlist
- Feature: Modo "aleatório" inteligente

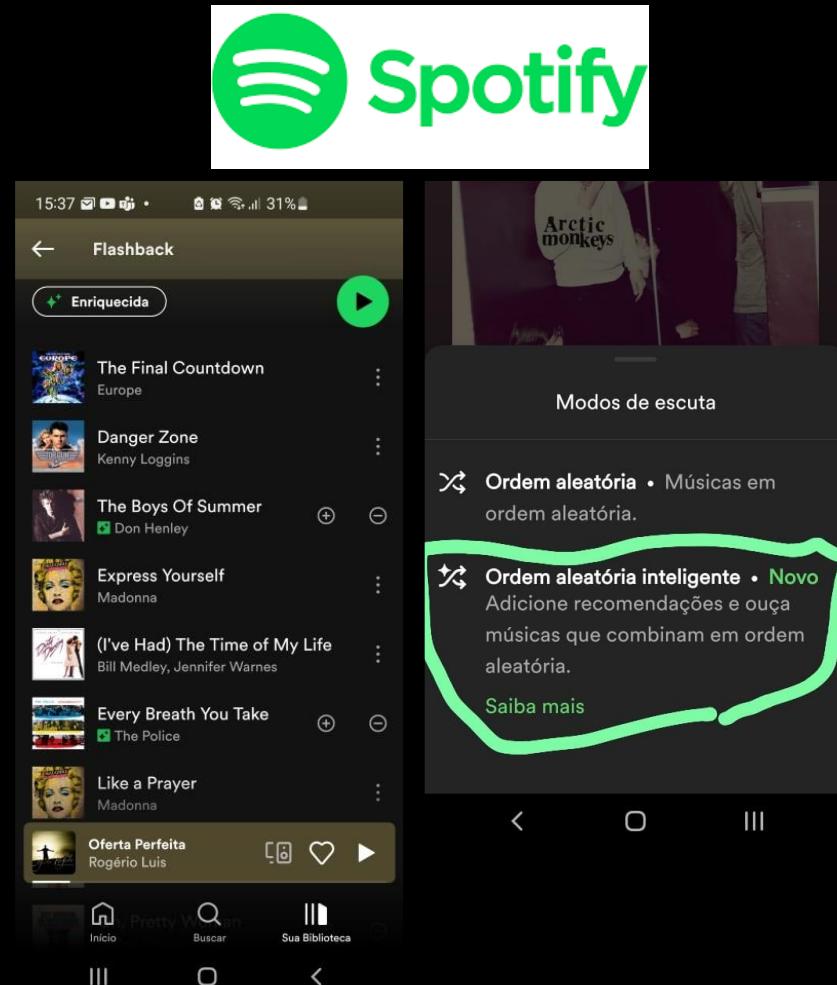
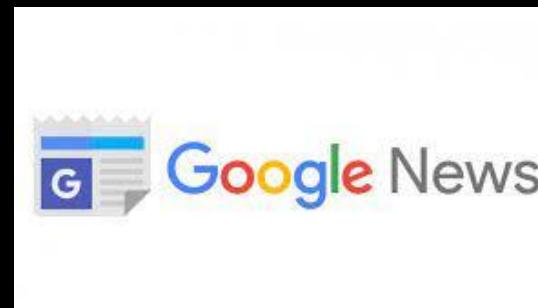


IMAGEM 10 - SPOTIFY

JORNAL, REVISTA, NOTÍCIAS, PORTAL DE CONTEÚDO



JORNAL, REVISTA, NOTÍCIAS, PORTAL DE CONTEÚDO

Caso George Floyd: tropas ocupam ruas dos EUA após protestos violentos contra o racismo



pressionou o pescoço da vítima com o joelho durante uma abordagem na última segunda-feira.



RELACIONADAS



De Obama a Kim Kardashian, personalidades protestam por George Floyd



A longa história de segregação e conflito racial em Minneapolis, onde George Floyd foi morto pela polícia



Nike lança manifesto contra o racismo e ganha apoio da Adidas



Caso George Floyd: tropas ocupam ruas dos EUA após protestos violentos contra o racismo



última segunda-feira.



RELACIONADAS



De Obama a Kim Kardashian, personalidades protestam por George Floyd



A longa história de segregação e conflito racial em Minneapolis, onde George Floyd foi morto pela polícia



Nike lança manifesto contra o racismo e ganha apoio da Adidas

Publicidade relacionada com similaridade => maior valor agregado ao negócio



MISCELÂNEA

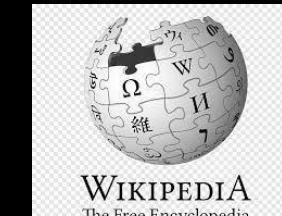
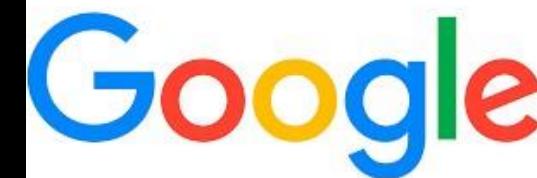
SOCIAL, CONTATOS, AMIZADES



VIAGENS, LAZER



PESQUISA, CONSULTA, BUSCA



Impactos dos SRs na sociedade

IMPACTOS

- Os SRs impactam a sociedade de forma geral: Consumo, preferências, amizades, relações, comportamento, carreira, lazer, eleições, tomadas de decisões em geral.



IMAGEM 14 - IMPACTOS

Capítulo 2

Propriedades e Problemas dos SRs

Capítulo 2

2.1. Propriedades

PROPRIEDADES

Aleatoriedade

escolher qualquer item para recomendar, possibilitando a combinação desta com outra propriedade

Similaridade

escolher itens com base na similaridade entre eles

Personalização do Usuário

modelagem de características, descoberta de preferência do usuário e histórico de interação do usuário com o sistema

PROPRIEDADES

Cobertura

proporção de itens recomendados pelo total de itens do seu estoque, visando explorar ao máximo sua base

Popularidade

escolher itens a recomendar com base em indicadores de consumo, por exemplo: volume de compras, cliques, compartilhamentos, visualizações e leituras.

Novidade

sugerir itens novos com base em data/hora de entrada deles no sistema

PROPRIEDADES

Diversidade

capacidade do recomendador em sugerir itens diversos, com base em distintos critérios e propriedades

Rankeamento

ordenar a lista de recomendação a partir do melhor item encontrado

Serendipidade

Trata-se do efeito de uma surpresa positiva causada por uma recomendação

Capítulo 2

2.2. Problemas

PROBLEMAS

Cold start

falta de capacidade do recomendador em sugerir itens por ausência de indicadores (itens ou usuários), problema encontrado no início do SR ou em novos itens e usuários

Aleatoriedade

escolher qualquer item para recomendar, dando a sensação ao usuário de que a recomendação é irrelevante

Previsibilidade

escolher sempre itens com base em um mesmo critério (exemplo: similaridade, conhecimento), podendo deixar o usuário entediado com o sistema

PROBLEMAS

Falta de conhecimento do usuário

ausência de características do usuário armazenadas pelo sistema e falta de histórico de interação do usuário com o sistema

Rankeamento

Não encontrar uma ordem boa para listar os itens, deixando itens piores no começo e piores no final da lista

Surpresa negativa

trata-se do efeito de uma surpresa negativa causada por uma recomendação, exemplo: um filme recomendado com base no perfil de outro usuário que exponha um dos usuários ou desgrade o outro

PROBLEMAS

Adicionalmente os SRs podem sofrer ataques maliciosos ou ainda tentativas de induzir o sistema com cliques massivos tornando o sistema enviesado. Exemplos:

- Cliques ou views massivos em redes sociais para alavancar um post ou perfil
- Votações massivas em enquetes com uso de robôs ou perfis fakes

PROBLEMAS

Exemplos

Notícia em leitura	Recomendação de notícia	Problema
Autor conta histórias de animais notívagos do bioma brasileiro	Nenhuma recomendação encontrada	Cold start
Trump acusa FBI de politizar investigações	Brasil deve ter 600 mil novos casos de câncer por ano	Aleatoriedade
Fake News são desafios para institutos de estatística	Conselho de comunicação cria comissão para avaliar projetos sobre fake news	Previsibilidade

Fonte: www.ebc.com.br

Capítulo 3

Técnicas, Algoritmos e Abordagens

Capítulo 3

3.1. Técnicas

TÉCNICAS

Recomendação baseada em conhecimento

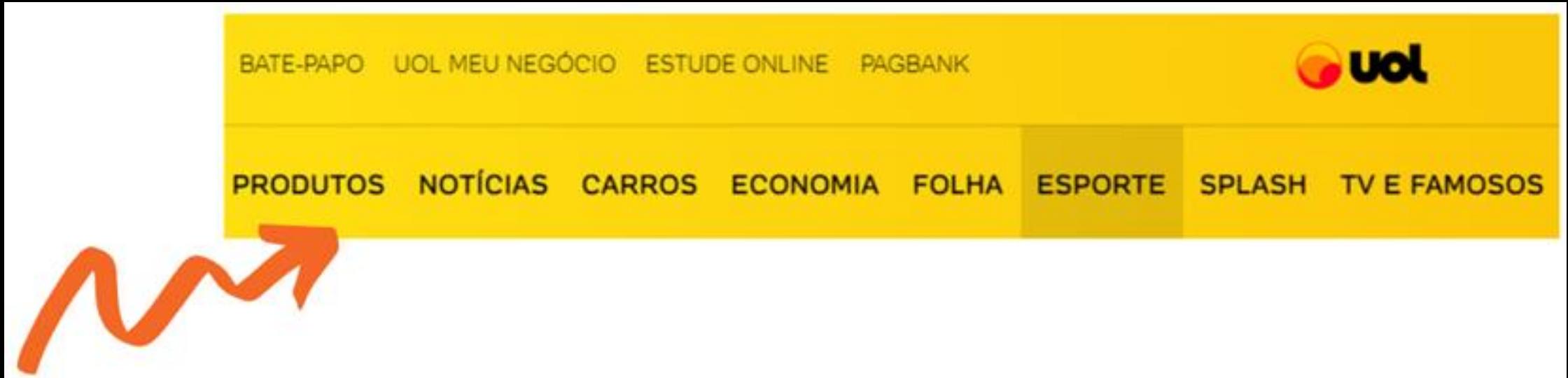


IMAGEM 15 - CONHECIMENTO

Considera o conhecimento específico (e a priori) a respeito do domínio e características dos itens.

TÉCNICAS

Recomendação baseada em conteúdo



IMAGEM 16 - CONTEÚDO

Considera palavras chaves ou o conteúdo textual de um item para ser usados como comparação e encontrar similaridade com outros itens.

TÉCNICAS

Pré-processamento de texto

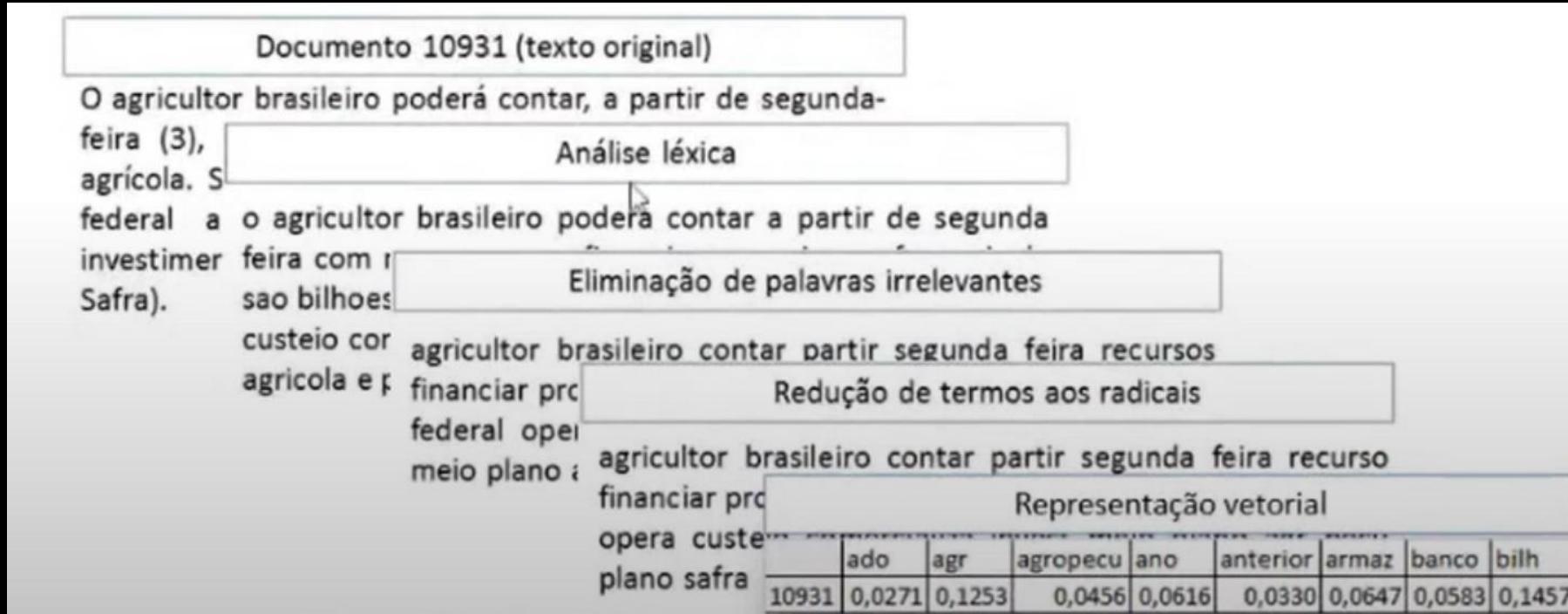


IMAGEM 17 – PRÉ-PROCESSAMENTO

=> Análise léxica (tokenizing) => Eliminação de palavras irrelevantes (stop words)
=> Redução dos termos aos radicais (stemming) => Representação vetorial

TÉCNICAS

FILTRO COLABORATIVO

- Recomenda itens a um usuário que outro usuário, similar a este, consumiu no passado.
- O termo consumiu depende da abordagem a ser adotada e pode representar uma compra, uma curtida, uma leitura ou uma música ouvida, por exemplo.



IMAGEM 18 – FILTRO COLABORATIVO

TÉCNICAS

Raciocínio baseado em casos e Recomendação baseada em casos



IMAGEM 19 - CASOS

Raciocínio baseado em casos considera a associação de um problema a uma solução (caso)

Recomendação baseada em casos é uma extensão do Raciocínio com modelagem que usa a recomendação baseada em conhecimento.

TÉCNICAS

Agrupamento de Dados

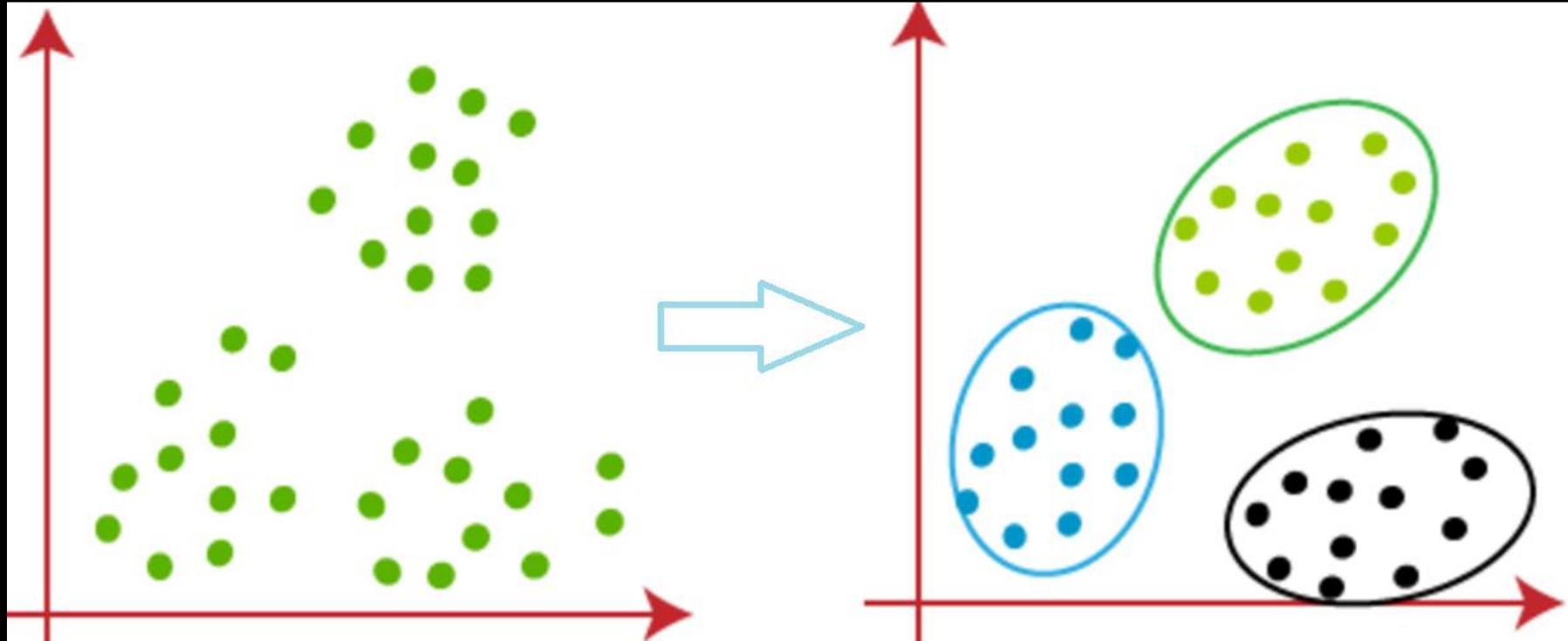


IMAGEM 20 - AGRUPAMENTO

Formar grupos sem conhecimento a priori, com base em similaridade e com objetivo de maximizar a similaridade intragrupo. Minimizar a similaridade intergrupo.

TÉCNICAS

Classificação

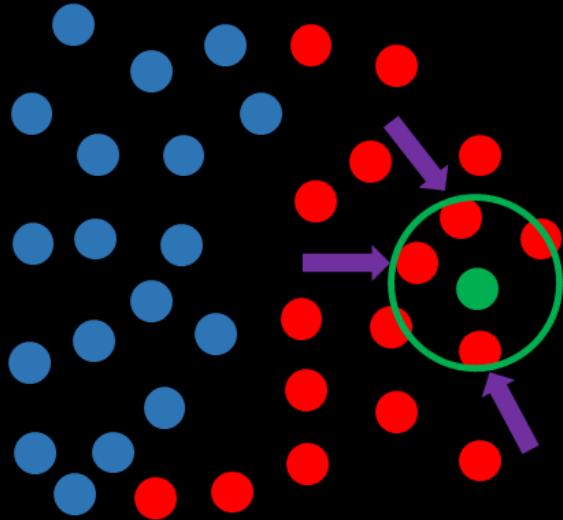


IMAGEM 22 - CLASSIFICAÇÃO

A classificação é uma técnica de aprendizagem supervisionada que usa dados rotulados com antecedência, com objetivo de fazer previsões futuras.

Isso quer dizer que treinamos os dados para que, dada a entrada de um novo item, esperamos “prever” em qual rótulo, grupo ou classe iremos classificar esse item.

TÉCNICAS

Regras de associação

ID	Pão	Leite	Fralda	Cerveja	Ovo	Café
1	1	1	0	0	0	0
2	1	0	1	1	1	0
3	0	1	1	1	0	1
4	1	1	1	1	0	0
5	1	1	1	0	0	1

IMAGEM 21 - ASSOCIAÇÃO

Modela os dados de forma que itens associados que pertençam a um mesmo sub-conjunto de uma base, se encontrados em uma dada ocorrência irão implicar na ocorrência de um outro item. Como exemplo, $\{leite, fralda\} \rightarrow \{cerveja\}$ é uma associação que diz que quando se encontra os itens *leite* e *fralda* em uma ocorrência, é esperado que a *cerveja* também apareça na transação.

Capítulo 3

3.2. Controles e Algoritmos

CONTROLES E ALGORITMOS

Podemos construir um SR implementando controles relativamente simples em termos de algoritmo:

Controle
Algoritmo



Controle
Algoritmo



CONTROLES E ALGORITMOS

Podemos tornar um SR mais sofisticado utilizando algoritmos um pouco mais complexos

Controle
Algoritmo

Preferências do usuário
{ DER, SQL }

Histórico do usuário
{ DER, SQL }

RB em conhecimento
{ DER, SQL }

Controle
Algoritmo

RB em conteúdo
{ Pipeline preProc, Dist}

Distância e Similaridade
{ Euclidiana, Cosseno, Sim}

Filtragem colaborativa
{ Sim, abordagem}

CONTROLES E ALGORITMOS

Podemos aperfeiçoar ainda mais um SR com uso de técnicas estudadas pela Inteligência artificial e algoritmos de datamining e machine learning, o que torna o desenvolvimento mais complexo

Controle

Algoritmo

Controle

Algoritmo

Raciocínio
Baseado em casos

{
CBR
}

RB em casos

{
CBR, DER,
SQL
}

Regras de
Associação

{
DER, SQL,
Apriori
}

Pré-
processamento

{ R (TM), Python
(NLTK) }

Agrupamento

{
K-Means
}

Classificação

{ K-NN, RNA}

RB = Recomendação Baseada

TÉCNICAS, CONTROLES E ALGORITMOS

Com tudo que vimos anteriormente é possível estabelecer uma associação para auxiliar no entendimento de que tipo de algoritmo pode ser aplicado a uma determinada técnica ou controle, desde os mais simples até os mais complexos e sofisticados.

Controles e Técnicas	Algorítmos
Clicks	: Evento on_click e incremento
Curtidas	: Evento on_click e incremento/decremento
Compartilhamentos	: Evento on_click e incremento
Tempo de Permanência	: Timer
Navegação do usuário	: Evento on_click e estruturas de dados do tipo lista, collections ou árvores
Busca por um filtro exato	Comando SQL Select com cláusula Where
Busca por Palavras chave	: Comando SQL Select com cláusula Where e like
Características do item	: Modelagem relacional e Selects com joins e Where
Preferências e histórico do usuário	: Modelagem relacional e Selects com agrupamento e soma
Recomendação baseada em conhecimento	: Modelagem relacional e Selects com joins e Where
Recomendação baseada em conteúdo	: Pipeline de Pré-processamento de Texto R (TM) e Python (NLTK)
Distância	: Fórmula matemática para cálculo de Distância, exemplo: Euclidiana e Cosseno
Similaridade	: Fórmula matemática inversamente proporcional à distância na escala entre 0 e 1
Filtro colaborativo	: Função calculada a partir da distância ou similaridade associada a uma abordagem de recomendação
Agrupamento	: Algoritmos de Clustering, exemplo K-means
Classificação	: Algoritmos de machine learning, exemplo K-NN e Redes Neurais Artificiais
Regras de associação	: Modelagem Relacional e algoritmo Apriori
Raciocínio e recomendação baseada em casos	: Metodologia CBR (Case based reasoning), Modelagem Relacional e recomendação baseada em conhecimento

Capítulo 3

3.3. Abordagens

ABORDAGENS

- A construção de um SR pode ser iniciada a partir do uso de uma técnica isolada de recomendação.
- Uma abordagem híbrida é considerada a partir do momento que se implementa duas ou mais técnicas e o sistema é capaz de calibrar o uso destas técnicas utilizando algum tipo de estratégia técnica ou de negócio.

ABORDAGENS

Exemplo de utilização de uma abordagem híbrida

Problema	Filtro colaborativo	Conteúdo	Conhecimento	Abordagem híbrida
<i>Cold start de item</i>	⬇️ ✗	⬆️ ✓	⬆️ ✓	⬆️
Previsibilidade	⬆️ ✓	⬇️ ✗	⬇️ ✗	⬆️
Conhecimento do usuário	⬆️ ✓	⬇️ ✗	⬆️ ✓	⬆️
Serendipidade	⬆️ ✓	⬇️ ✗	⬇️ ✗	⬆️

IMAGEM 22 - CLASSIFICAÇÃO

No limite, a abordagem híbrida pelo menos empata com a eficácia de uma técnica isolada, mas ao longo do tempo, na média, o uso da abordagem híbrida tende a ter uma eficácia maior em relação à uma técnica individual.

TÉCNICAS, ALGORITMOS E ABORDAGENS

Conclusões

- É um consenso entre os estudiosos que o uso de uma propriedade isolada para sugerir itens, já define um SR
- Algumas propriedades, se usadas de forma isoladas ou fora de contexto, podem trazer problemas ao SR
- Para atacar um problema, implementamos outra propriedade que também pode trazer seus problemas específicos

TÉCNICAS, ALGORITMOS E ABORDAGENS

Conclusões

- O uso de uma abordagem híbrida tende a minimizar os problemas e melhorar alguns aspectos da recomendação, dessa forma o uso do mix apropriado de propriedades é o que define um bom recomendador. Este tipo de abordagem traz consigo um aumento no esforço de desenvolvimento, na complexidade do algoritmo e maior volume de dados e transações podendo acarretar dificuldades de performance.
- Adicionalmente, seu SR pode implementar algoritmos que auxiliem na ponderação e calibragem do uso das propriedades, fazendo com que seu sistema se torne mais inteligente ao longo do tempo.

Capítulo 4

Estudos de Casos de Sistemas de Recomendação

Capítulo 4

4.1. Recomendação na área financeira + Estudo de
Caso Open Finance

Estudo de Caso: Área financeira utilizando Open Finance

Desafios dos SRs para área financeira: Enquanto SRs de domínios mais tradicionais, como os vistos anteriormente, privilegiam ações mais imediatas, na área financeira há questões que envolvem:

- Comprometimento de longo prazo
- Impacto financeiro da recomendação na vida do usuário
- Mudança de natureza dos produtos ao longo do tempo
- Regulamentações
- Moedas
- Poucos indicadores de feedbacks dos usuários
- Privacidade e leis de proteção de dados (risco de cold-start).

Estudo de Caso: Área financeira utilizando Open Finance

Objetivo: Apresentar uma proposta de arquitetura e um protótipo de um SR utilizando Open Finance para clientes Pessoa Física do banco Santander.

O que é o Open Finance?

Open Finance é a liberdade de compartilhar seus dados financeiros. Que dados? Ora, todo o seu histórico financeiro até hoje. Informações sobre as **contas que já pagou**, o **tipo de cartão** que você tem, seus **limites disponíveis**, os **emprestimos realizados**, **produtos contratados** e muito mais. Agora, sabe por que isso é legal? Você aumenta seu poder de escolha.

Ao conhecer seus dados, as instituições financeiras de sua escolha poderão apresentar a você propostas mais atraentes, como oferecer **serviços extras na conta corrente**, **isenções em cartão**, **taxas menores no crédito** ou **condições melhores para renovar um seguro**, por exemplo.



IMAGEM 31 –OPEN FINANCE O QUE É

O Estudo de caso iniciou acessando o portal aberto do Santander e pesquisando sobre Open Finance e quais dados poderiam ser compartilhados para serem usados em um SR.

Estudo de Caso: Área financeira utilizando Open Finance

Quais outras informações relevantes para montagem da arquitetura e do protótipo?



The image shows a smartphone displaying the "Open Finance Santander" application. The screen is red and features the Santander logo at the top. Below it, there's a section for "Caixa Econômica Federal" with a blue circular icon and the text "Atualizado em 30 de mar.". A "Contas" section follows, showing an account entry for "Ag 8501 C/C 9501-5".

Flexibilidade

Em breve, você poderá movimentar todas as suas contas, por meio dos canais digitais do Santander, mesmo que sejam de outra instituição financeira, a qualquer hora e lugar. Os nossos [canais digitais](#) estarão preparados para você consultar saldo, extrato, fatura, realizar investimentos e até fazer pagamentos, e, o melhor, com toda segurança do Banco Santander.

Como será o compartilhamento de dados entre as instituições financeiras?

As instituições devem [disponibilizar as informações por meio de interfaces digitais \(APIs\)](#).

O que é API?

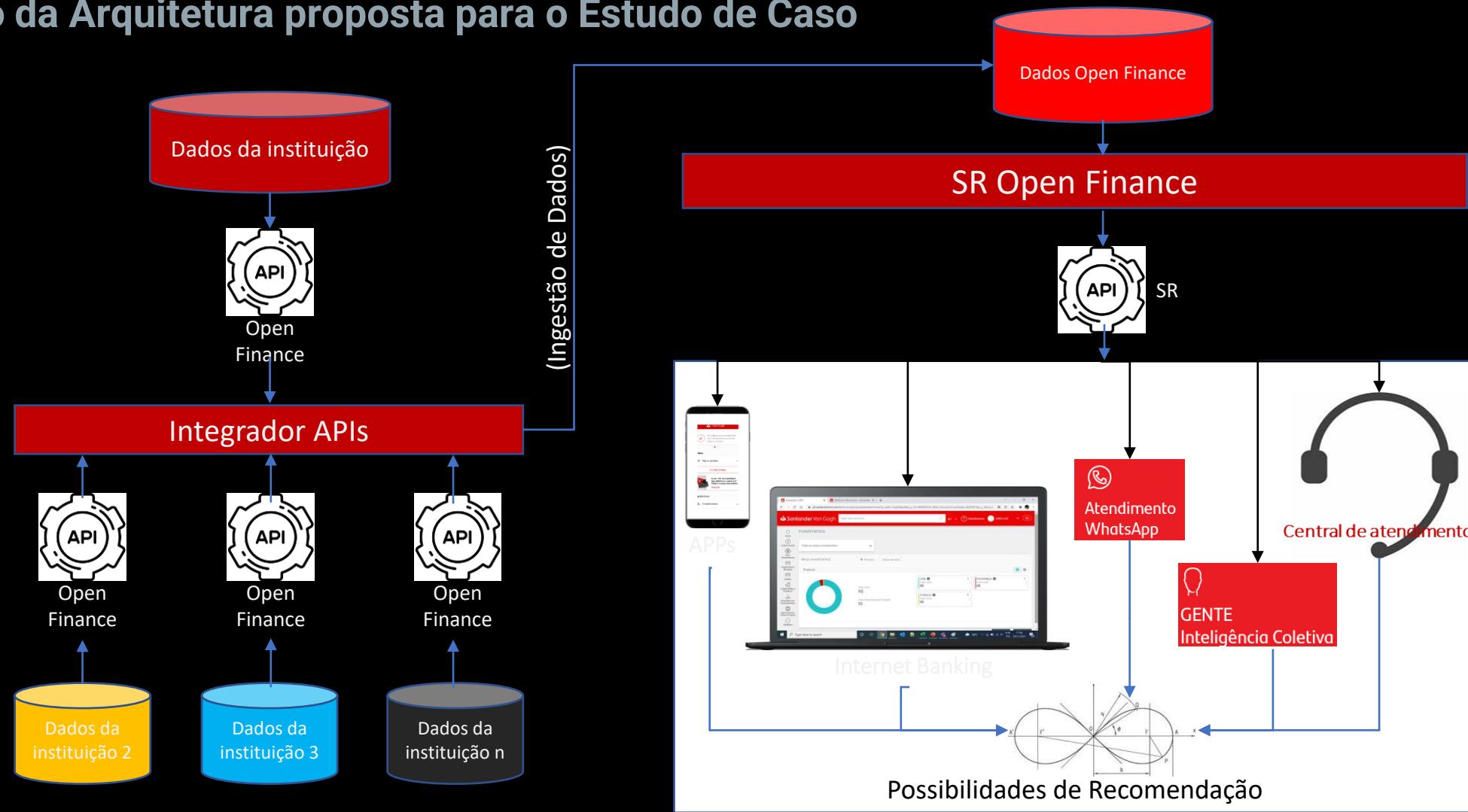
API é a sigla para “Application Programming Interface”. É uma espécie de ponte que conecta diferentes plataformas a uma mesma base. Ou seja, quando os dados passam a ser compartilhados, as instituições autorizadas conseguem trocar informações entre si de maneira segura e padronizada, seguindo as regras e parâmetros estabelecidos pelo órgão regulador.

IMAGEM 32 –OPEN FINANCE CANAIS E API

Importante considerar a disponibilização do SR nos diferentes canais do banco e utilizar o padrão API para integração e comunicação entre as instituições.

Estudo de Caso: Área financeira utilizando Open Finance

Desenho da Arquitetura proposta para o Estudo de Caso



Estudo de Caso: Área financeira utilizando Open Finance

Protótipo – Cenário 1

- Cliente Silvio Ramos da Instituição 2 compartilhou dados Open Finance com o Santander
- O cliente já possuía uma conta no Santander
- Na Instituição 2, o cliente possui um cartão de crédito com as seguintes características
 - Limite de crédito de R\$ 3000,00
 - Taxa de Juros Rotativa de 10% ao mês
 - Programa de pontuação de 1 ponto a cada 2 reais gastos
 - Anuidade no valor de 120,00
- O cliente acessa o Internet Banking do Santander
- Como o SR Open Finance poderia oferecer a melhor recomendação a esse cliente?

Estudo de Caso: Área financeira utilizando Open Finance

Protótipo – Cenário 1 – Desenho proposto

The screenshot shows the Santander Van Gogh Open Finance prototype interface. The top navigation bar includes the Santander Van Gogh logo, a search bar with placeholder "Digite aqui sua busca", and user profile options for "Silvio Ramos". The main content area features several cards:

- Conta Corrente**: Shows a circular icon with a dollar sign and arrows. Text: "Tenha o App Santander no seu celular e acompanhe sua conta de onde estiver." Below: "Consulte aqui a posição consolidada do seu saldo da conta corrente e seu limite contratado." Includes an "Exibir" button.
- Cartão de Crédito**: Shows a credit card icon. Text: "Você tem um cartão de crédito pré-aprovado !!! Vantagens do Cartão de Crédito Santander:
 - Limite de crédito de R\$ 4000,00
 - Taxa de Juros Rotativa de 9% ao mês
 - Programa Esfera: 1 ponto a cada 1 real gastos
 - Isento de Anuidade" Includes an "Exibir" button and a "Solicitar" button.
- Investimentos**: Shows a circular icon with a clock and coins. Text: "Consulte aqui a posição consolidada de seus Investimentos." Includes an "Exibir" button.
- Empréstimos**: Shows a circular icon with a hand holding a coin. Text: "Consulte aqui seu limite de empréstimo pré aprovado e aproveite para contratar." Includes an "Exibir" button.
- Favoritos**: Shows a list of items: "Transferências" (with "Programar transfe..."), "Pagamentos" (with "Pagar títulos e con..."), "Cartões" (with "Cadastrar Bônus E..."), and "Utilidades" (with "2º via de comprov...").
- Callout**: A large blue callout bubble covers the "Cartão de Crédito" and "Investimentos" cards. It contains the text: "Você sabe qual é o seu perfil investidor? Se ainda não sabe, preencha o API (Análise de Perfil do Investidor)." and a "PREENCHA AQUI!" button.

A sidebar on the left lists various financial categories with icons:

- Shopping
- Conta Corrente
- Pix e Transferências
- Pagamentos e Recargas
- Cartões
- Investimentos e Poupança
- Empréstimos e Financiamentos
- Open Finance e Outros Produtos
- Utilidades

IMAGEM 34 –OPEN FINANCE PROTOTIPO 1

Estudo de Caso: Área financeira utilizando Open Finance

Protótipo – Cenário 2

- Cliente Silvia Regina da Instituição 3 compartilhou dados Open Finance com o Santander
- A cliente já possui no Santander
- Na Instituição 3, a cliente possui uma carteira de investimentos a seguir:
 - R\$ 3000,00 na Poupança
 - R\$ 10000,00 no CDB DI 80% do CDI
 - R\$ 2000,00 em um Fundo de ações das empresas mais recomendadas da bolsa
 - Perfil do investidor: Moderado
- Saldo em conta corrente na Instituição 3: R\$ 8000,00
- O cliente acessa o App do Santander
- Como o SR Open Finance poderia oferecer a melhor recomendação a esse cliente?

Estudo de Caso: Área financeira utilizando Open Finance

Protótipo – Cenário 2 – Desenho proposto

Investimentos

Todas as contas e investimentos

Meus investimentos Produtos Classe de ativos

Produtos

Carteira recomendada pelo SR Open Finance

Simulação de retorno em 1 ano: R\$ 2.000,00
Performance 33% melhor que sua carteira atual

The screenshot shows a user interface for managing investments. At the top, there's a dropdown menu set to 'Todas as contas e investimentos'. Below it, a section for 'Meus investimentos' has two radio button options: 'Produtos' (selected) and 'Classe de ativos'. A large circular donut chart on the left indicates the distribution of investments across different asset classes. To the right, a section titled 'Carteira recomendada pelo SR Open Finance' displays four investment products: 'CDB DI integrado C/C' (Valor total: 80% do CDI, R\$ 8.000), 'POUPANÇA' (Valor total: R\$ 1.000), 'CDB Pré-fixado 110%' (Valor total: CDI, R\$ 11.000), and 'FUNDOS' (Carteira Valor Algo+, Valor total: R\$ 3.000). At the bottom, a callout box highlights a performance simulation: 'Simulação de retorno em 1 ano: R\$ 2.000,00' and 'Performance 33% melhor que sua carteira atual'.

Estudo de Caso: Área financeira utilizando Open Finance

Insights sobre recomendação na área financeira

- **Previdência:** PGBL ou VGBL, tipo de tarifa, adicionar cobertura por morte, invalidez, fundos de renda fixa, variável, mix de fundos.
- **Seguro:** Adicionar serviços de assistência, guincho, franquia de para-brisa, carro reserva.
- **Carteira de investimento:** Sugerir investimentos com base em: Perfil de investidor, histórico de investimentos, patrimônio, outros produtos contratados.
- **Empréstimos e financiamentos:** Sugerir crédito com base em: Scoring, histórico do cliente, renda e comprometimento da renda, produtos contratados, saldo em conta corrente, investimentos, uso do cartão de crédito.

Estudo de Caso: Área financeira utilizando Open Finance

Insights sobre recomendação na área financeira

- **Cartões de Crédito:** Sugerir cartões considerando: Programa de benefícios, limites, histórico, uso internacional, uso corporativo, exclusividade em pré-vendas.
- **Onboarding** (Cadastro de novos clientes): Reaproveitar dados cadastrais do seu cliente de outra instituição de forma que ele possa se cadastrar na sua instituição sem precisar preencher formulários longos, facilitando a criação de contas digitais e simplificando a jornada de aprovação de diferentes produtos ofertados.

Estudo de Caso: Área financeira utilizando Open Finance

Conclusão sobre recomendação na área financeira

- Há muitas oportunidades de desenvolvimento para Sistemas de Recomendação em Instituições Financeiras, ainda mais com o Open Finance que possibilita obter dados do cliente de outras instituições.
- No entanto há também muitas possibilidades de coleta de novos dados e indicadores de pessoas e empresas que já são clientes da instituição para utilização em SR.
- Com todos os dados que a instituição já possui do cliente ou ainda pode coletar, associados com novos dados vindos do Open Finance, imaginou quantas possibilidades para desenvolvimento e evolução dos SRs na área financeira?

Capítulo 4

4.2. Material de Apoio - Estudo de Caso: Pesquisa
Acadêmica – Uma Abordagem Híbrida para Sistemas
de Recomendação de Notícias

Estudo de caso de um SR de notícias

Objetivo:

Construir um Sistema de Recomendação de notícias utilizando uma abordagem híbrida (mix de técnicas), conhecendo as preferências e o histórico do usuário, aplicando uma métrica original de ranqueamento e validando esse sistema por meio de um experimento online com participação de usuários reais.

Estudo de caso de um SR de notícias

Hipóteses:

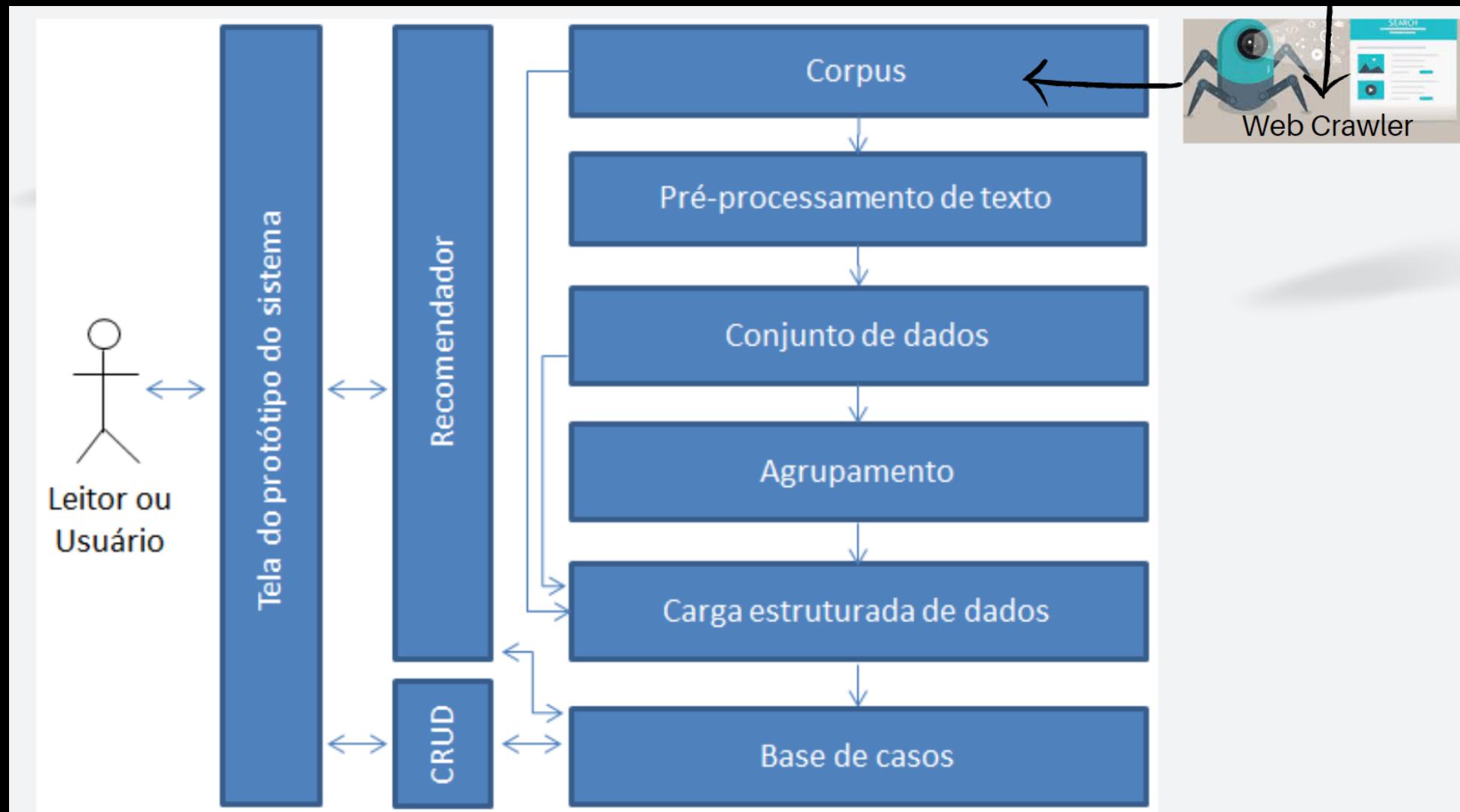
**Se (adotar uma abordagem híbrida) então
é possível equilibrar limitações do uso de métodos isolados**

**Se (conhecer o histórico e as preferências dos usuários) então
é possível aumentar a satisfação dos usuários**

**Se (implementar uma fórmula original de ranqueamento) então
é possível encontrar itens mais relevantes aos usuários**

Estudo de caso de um SR de notícias

Arquitetura



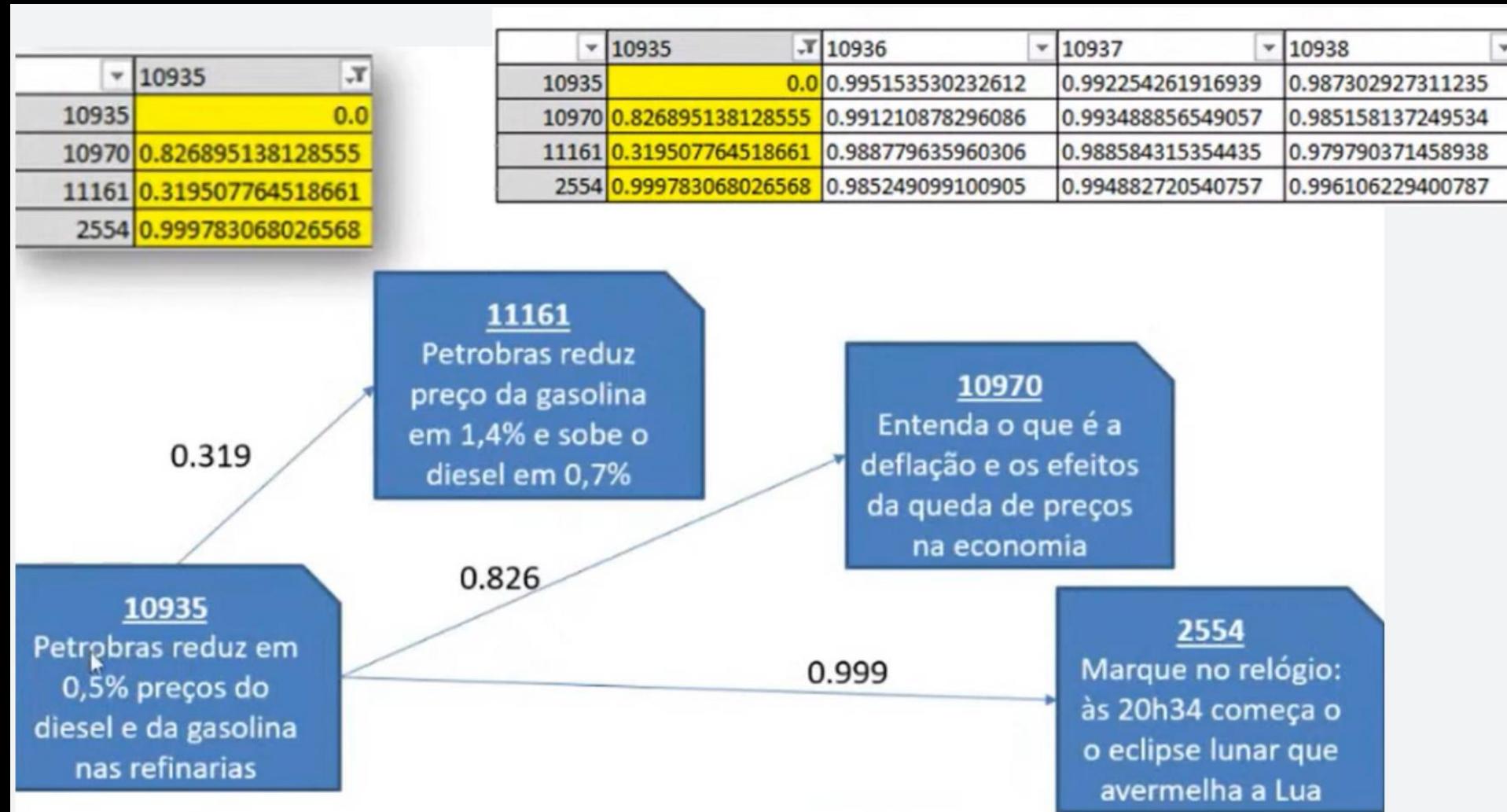
Estudo de caso de um SR de notícias

Pré-processamento de texto:

- Análise léxica (tokenizing): separador dos termos, tratamento de: pontuação, acento, maiúscula/minúscula.
- Eliminação de palavras irrelevantes (stop words): remover artigos, preposições, pronomes, numerais, advérbios.
- Redução dos termos aos radicais (stemming): retirar plural, gerúndio, aumentativo, diminutivo.
- Representação vetorial: criação de uma matriz baseada na frequência dos termos do corpus.

Estudo de caso de um SR de notícias

Do Pré-processamento de texto para a Matriz de Distância



Estudo de caso de um SR de notícias

Do pre-processamento de texto para o Agrupamento de Dados

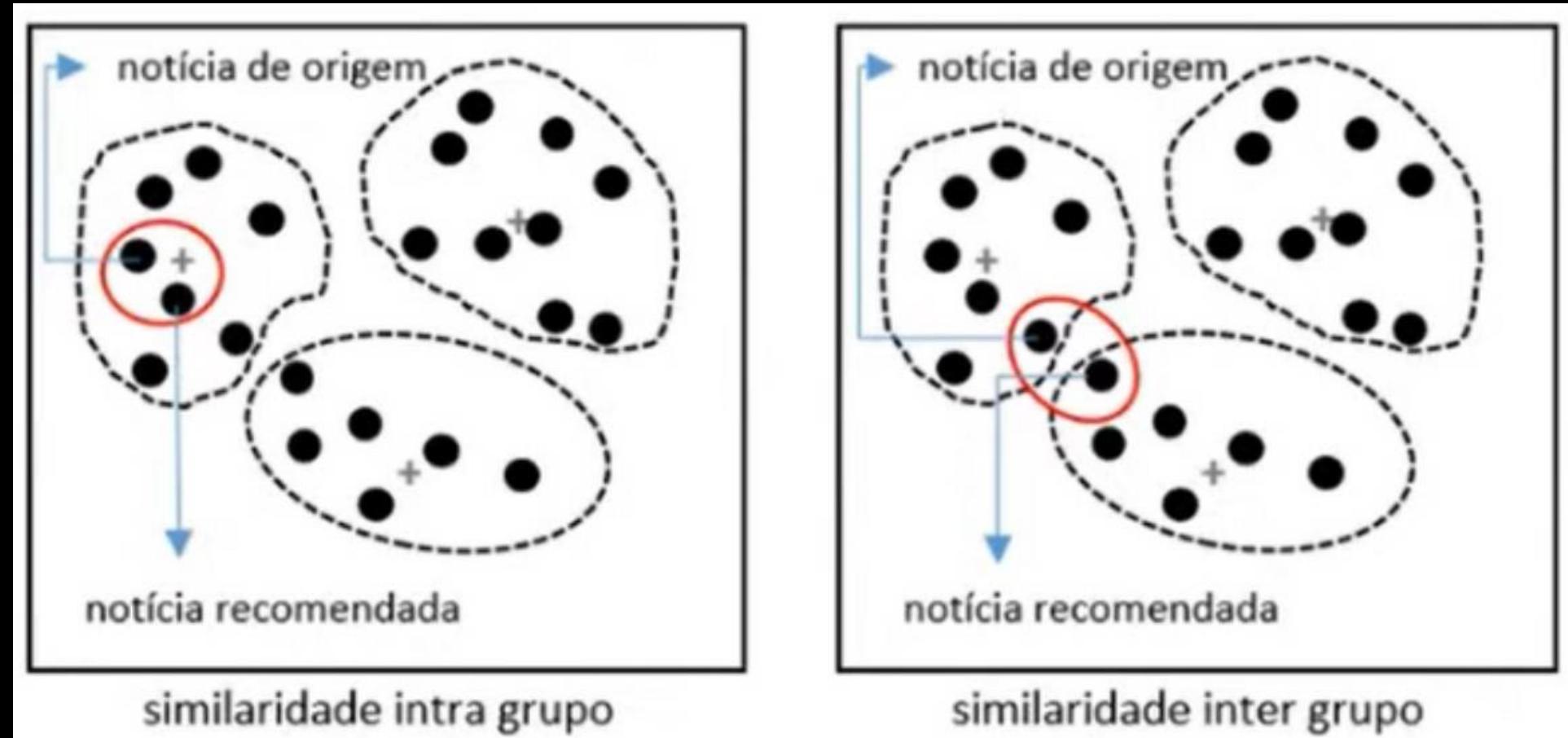


IMAGEM 25 – Agrupamento Dados

Estudo de caso de um SR de notícias

Base de Casos implementada por meio de modelagem relacional



IMAGEM 26 – Base de Casos

Estudo de caso de um SR de notícias

Interface gráfica do protótipo construído como portal de conteúdo para o experimento

[Protótipo de portal de notícias V1.0.3 - Experimento do projeto de pesquisa \(Pagnossim, J. L. M., 2017\)](#)
Usuário: Leitor anônimo ([Cadastrar um novo usuário](#)) [Sair do experimento](#)

Universo: Descoberta de planetas parecidos com a Terra desafia a ciência

Planeta Terra é rochoso, orbita uma estrela chamada sol, tem água em estado líquido, e tem vida. Essas são características que fazem a Terra se diferenciar dos outros planetas do nosso sistema solar, não é mesmo? É...Mas e de outros sistemas solares? Será que existem outras Terras? Olha, 2017 tem sido um ano marcante em relação às descobertas dos exoplanetas - são aqueles encontrados fora do nosso sistema solar. Mas, além disso, da descoberta de exoplanetas, o mais interessante é a descoberta de planetas muito parecidos com o nosso: são os chamados irmãos e até primos da Terra. De fevereiro para cá, dois momentos importantes para a astronomia. O primeiro foi o anúncio feito pela Nasa de que cientistas europeus encontraram em um único sistema solar sete planetas orbitando um tipo de estrela chamada de anã-vermelha. Destes sete, a maioria é rochosa e três têm altas chances de água na superfície. Tudo indica que eles podem ser muito parecidos com a Terra, tanto em relação ao tamanho, quanto a temperatura. E o segundo momento, agora em junho, foi o registro feito pelo telescópio Kepler: Mais de 200 planetas candidatos, sendo 10 planetas parecidos com a Terra. Quem explica um pouco melhor pra gente é o

Avaliação da notícia

 Clique neste ícone se curtiu esta notícia

Avaliação da recomendação

 Clique neste ícone se ficou surpreso positivamente com a recomendação desta notícia

 Forneça uma nota para a recomendação desta notícia

[Exibir notícias recomendadas](#)

Notícias obtidas do portal [EBC](#) (Empresa Brasil de Comunicação), mediante autorização prévia e indicação da fonte. ©

Estudo de caso de um SR de notícias

Propriedades, problemas, abordagem e estratégias para recomendação

Estratégia	Problema associado	Critérios combinados	Estratégia	Contexto da utilização
$\{ES_1\}$	<i>Cold start</i> de item	$\{CR_1\}, \{CR_2\}, \{CR_3\}$ e $\{CR_4\}$	$\{ES_4\}$	Existe uma solução conhecida para o problema
$\{ES_2\}$	Previsibilidade e aleatoriedade	$\{CR_5, \dots, CR_{11}\}$	$\{ES_5\}$	Trata-se de um novo problema, busca-se uma solução adaptada
$\{ES_3\}$	Falta de personalização de usuário	$\{CR_{12}\}$ e $\{CR_{13}\}$		

Critério	Descrição
$\{CR_1\}$	Recupera \mathcal{ND} com maior <i>sim</i> de conteúdo em relação à \mathcal{NO}
$\{CR_2\}$	Recupera a \mathcal{N} com maior <i>sim</i> intergrupo
$\{CR_3\}$	Recupera a \mathcal{N} com maior <i>sim</i> intragrupo
$\{CR_4\}$	Recupera uma \mathcal{N} aleatória que não tenha sido recomendada anteriormente
$\{CR_5\}$	Recupera a \mathcal{N} mais lida dentro do canal
$\{CR_6\}$	Recupera a \mathcal{N} mais lida do portal
$\{CR_7\}$	Recupera a \mathcal{N} mais curtida do canal
$\{CR_8\}$	Recupera a \mathcal{N} mais curtida do portal
$\{CR_9\}$	Recupera \mathcal{ND} com mais recomendações aceitas em relação à \mathcal{NO}
$\{CR_{10}\}$	Recupera \mathcal{ND} com maior nota de avaliação em relação à \mathcal{NO}
$\{CR_{11}\}$	Recupera \mathcal{ND} com maior indicador de serendipidade em relação à \mathcal{NO}
$\{CR_{12}\}$	Recupera a última \mathcal{N} lida pelo \mathcal{U} mais similar
$\{CR_{13}\}$	Recupera a \mathcal{N} mais lida do \mathcal{CFU} corrente

Estudo de caso de um SR de notícias

Rankeamento – matriz de competição

Fórmula original MURR (Métrica unificada para ranqueamento da recomendação)

	CR1	CR2	CR3	CR4	CR5	CR6	CR7	CR8	CR9	CR10	CR11	CR12	CR13	MURR	Posição
N1	1	2	2	2	12	2	6	12	5	5	12	8	13	82	1
N12	11	4	4	11	4	11	11	4	4	3	2	1	13	83	2
N4	5	9	11	1	2	7	5	2	8	8	9	5	13	85	3
N5	4	10	9	4	1	4	4	7	9	9	8	4	13	86	4
N9	9	5	5	9	5	9	9	5	1	2	4	11	13	87	5
N2	2	1	12	3	11	3	2	11	6	6	11	7	13	88	6
N11	12	3	3	12	3	12	12	3	3	4	1	9	13	90	7
N3	3	12	1	5	10	5	3	10	7	7	10	6	13	92	8
N10	10	6	6	10	6	10	10	6	2	1	3	10	13	93	9
N7	6	8	8	6	8	6	1	8	11	11	6	2	13	94	10
N6	7	11	10	7	9	1	7	9	10	10	7	3	13	104	11
N8	8	7	7	8	7	8	8	1	12	12	5	12	13	108	12
N13	13	13	13	13	13	13	13	13	13	13	13	13	1	157	13

Estudo de caso de um SR de notícias

Dashboard de resultados

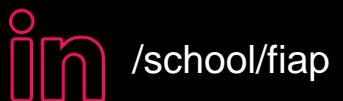


IMAGEM 30 – Resultados

MUITO OBRIGADO
E não deixe de realizar a survey
desta aula no portal FIAP.

 profjose.pagnossim@fiap.com.br

linkedin.com/in/jose-luiz-maturana-pagnossim-3454898



FIAP

Copyright © 2022 | Professor José Pagnossim

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente
proibido sem consentimento formal, por escrito, do professor/autor.

Capítulo 5

Desenvolvendo um Sistema de Recomendação

Capítulo 5

5.1. Configuração do Ambiente Python (IDLE) e criação
do Data set

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

Instalação do Python

- Se não tiver o Python instalado, abrir a página de downloads do Python.org (<https://www.python.org/downloads/>)
- Baixe a versão mais atual para seu Sistema Operacional
- Para os algoritmos que iremos implementar usaremos o Python 3.10.1 para Windows.

Para saber se vc já tem o Python instalado na máquina

- Abra o prompt de comando (Cmd)
- Digite `python --version`

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

- Instalação do Python
- Ao concluir o Download Python 3.10.1 para Windows, execute o arquivo “python-3.10.1-amd64.exe” baixado em seu computador, de preferência execute o arquivo em modo administrador

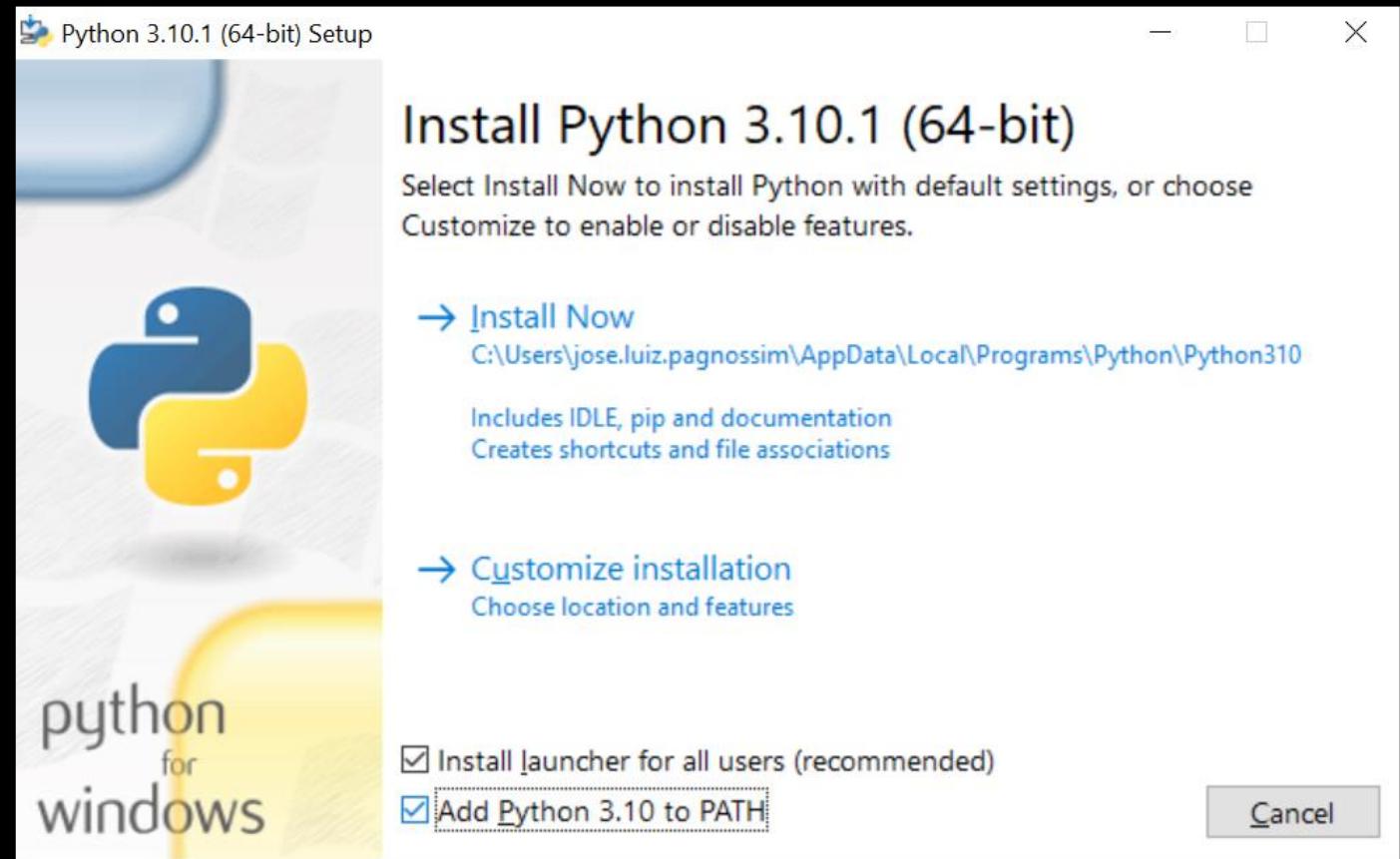


IMAGEM 36 –INSTALAÇÃO PYTHON 1

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

- Instalação do Python
- Escolha a opção Customize caso queira alterar alguma configuração na instalação ou Install Now se quiser a instalação padrão. Clique em Next

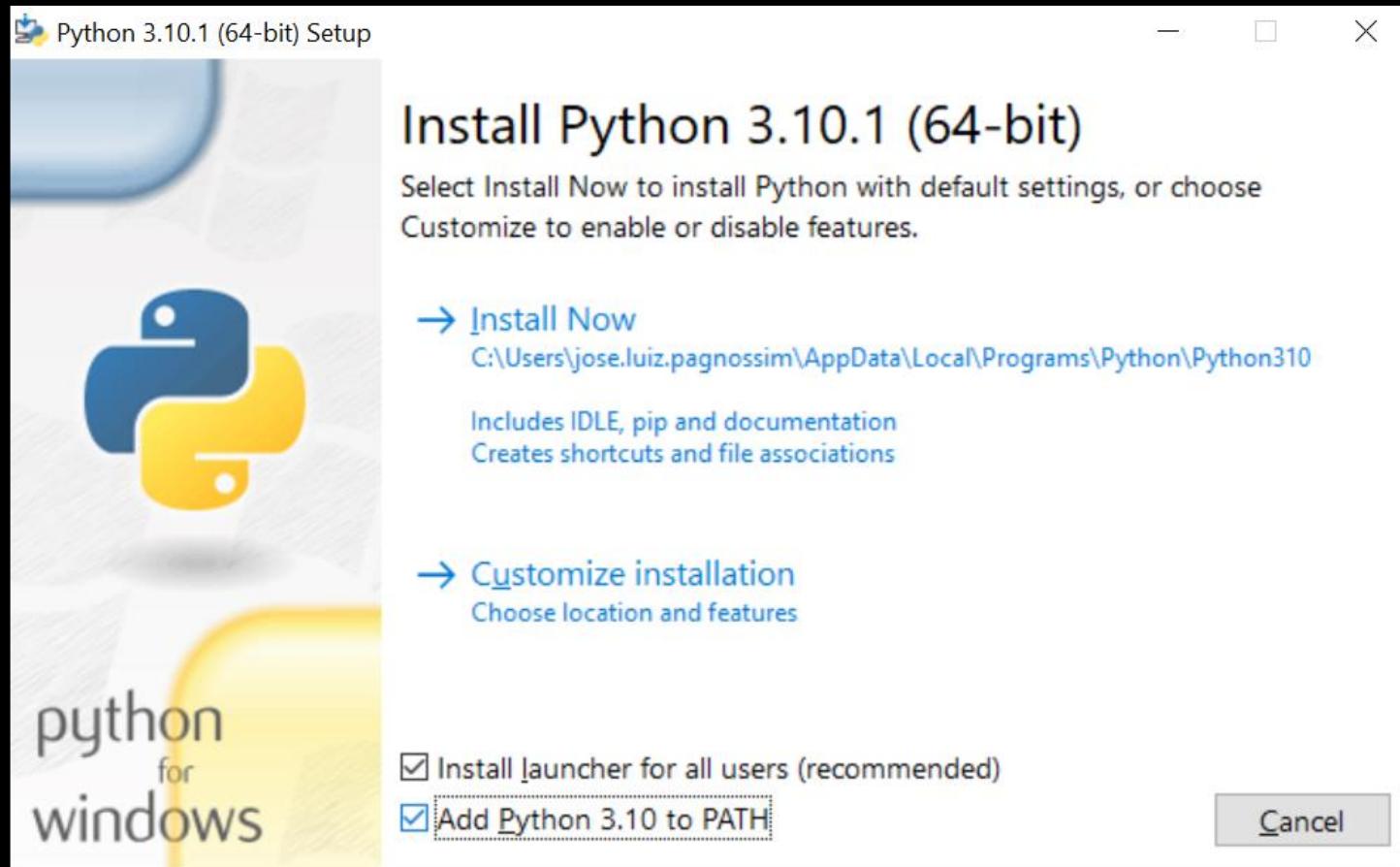


IMAGEM 37 –INSTALAÇÃO PYTHON 2

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

- Instalação do Python
- Marque ou desmarque opções como preferir e clique em Install

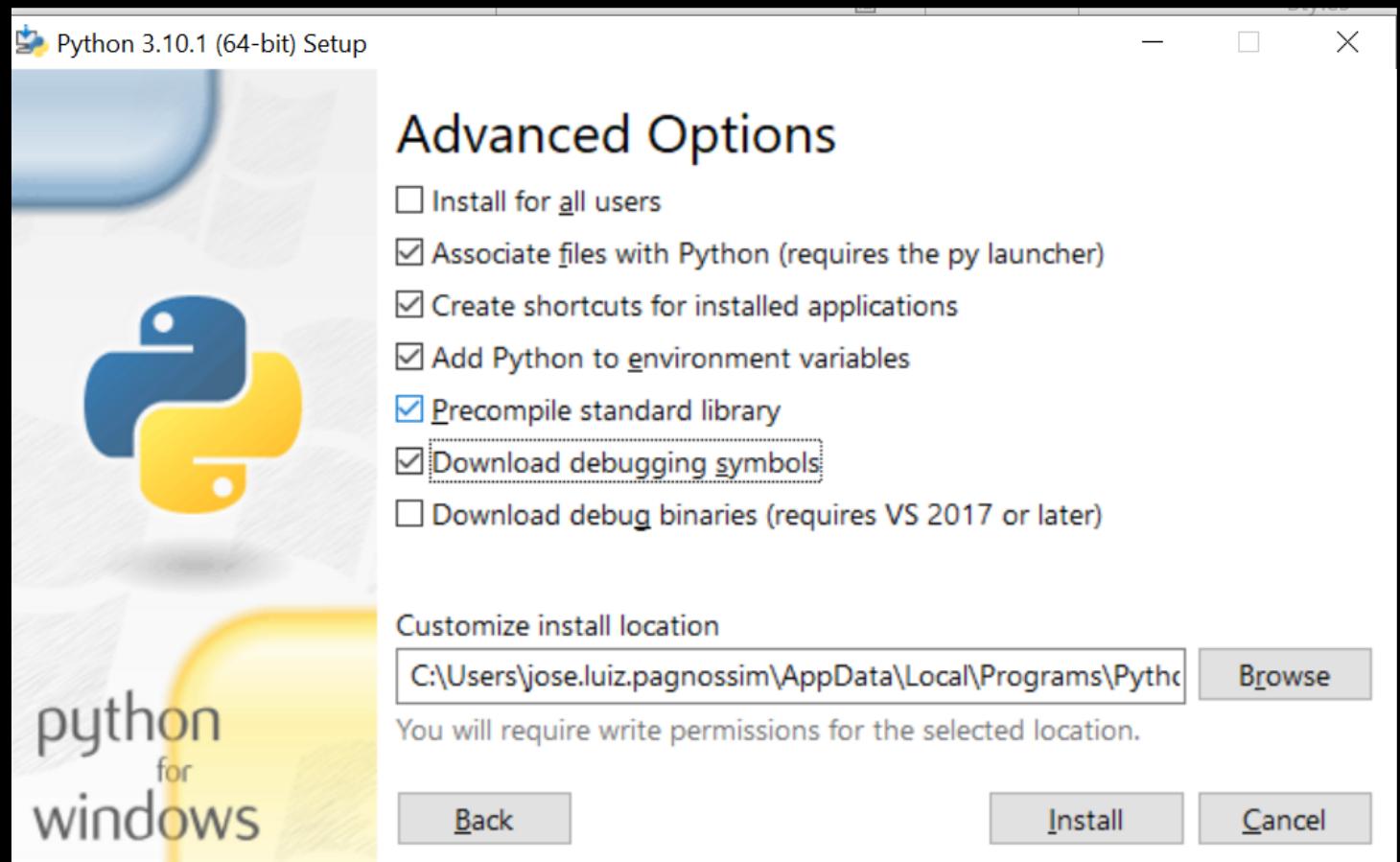


IMAGEM 38 –INSTALAÇÃO PYTHON 3

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

- Instalação do Python
- Será exibida uma barra de progresso até a conclusão. Pode ocorrer do instalador solicitar para você desabilitar o limite de tamanho no path. Se desejar deabilitar, clique na opção “Disable path length limit” e por fim clique em close para fechar o instalador.

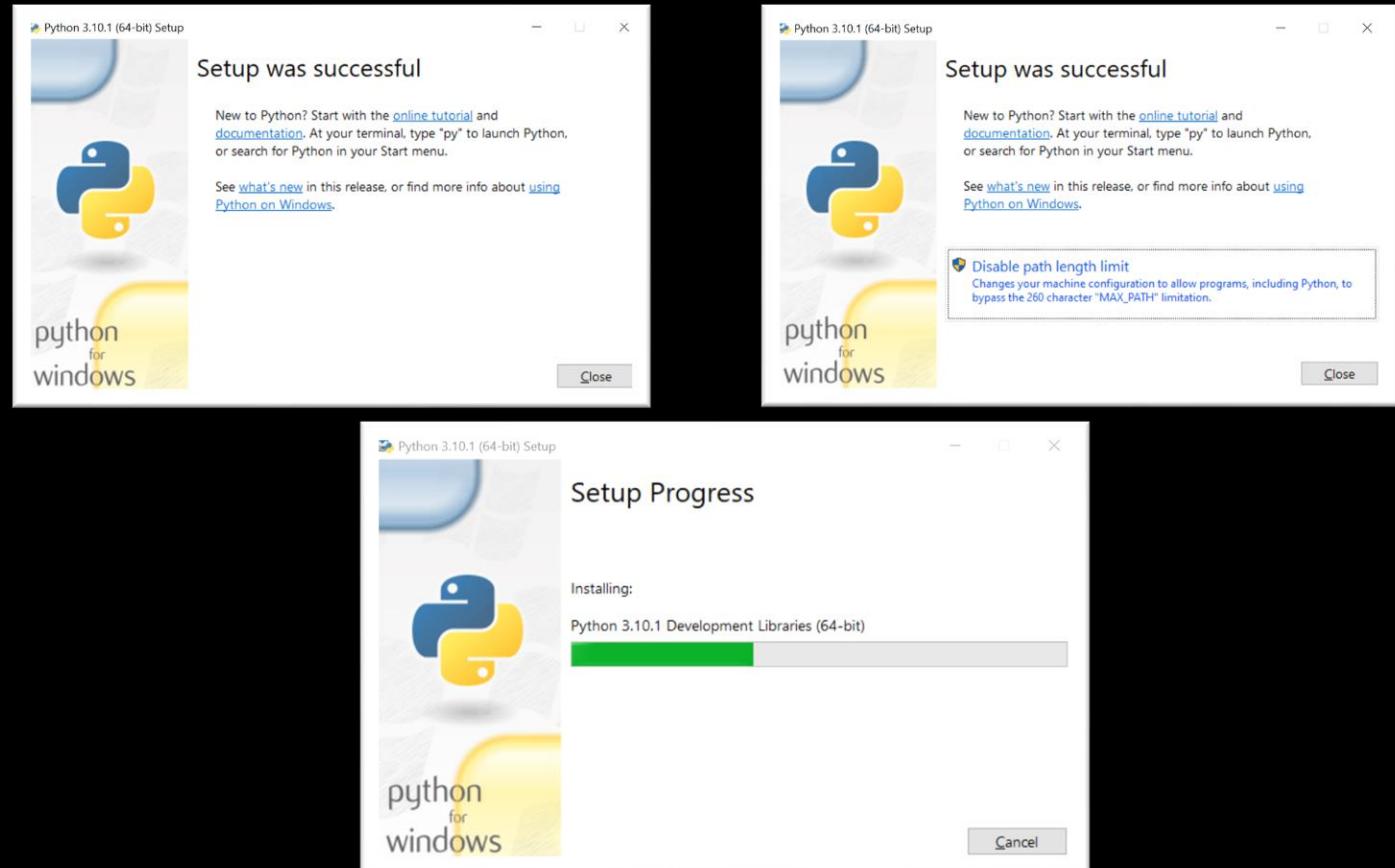


IMAGEM 39 –INSTALAÇÃO PYTHON 4

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

- Ambiente de desenvolvimento IDLE
- Após a instalação, procure no computador por “IDLE”
- O IDLE (Ambiente de Desenvolvimento e Aprendizagem Integrado) é um ambiente de desenvolvimento integrado (IDE) para Python. O instalador do Python para Windows contém o módulo IDLE por padrão.
- <https://python.org.br/installacao-windows/>

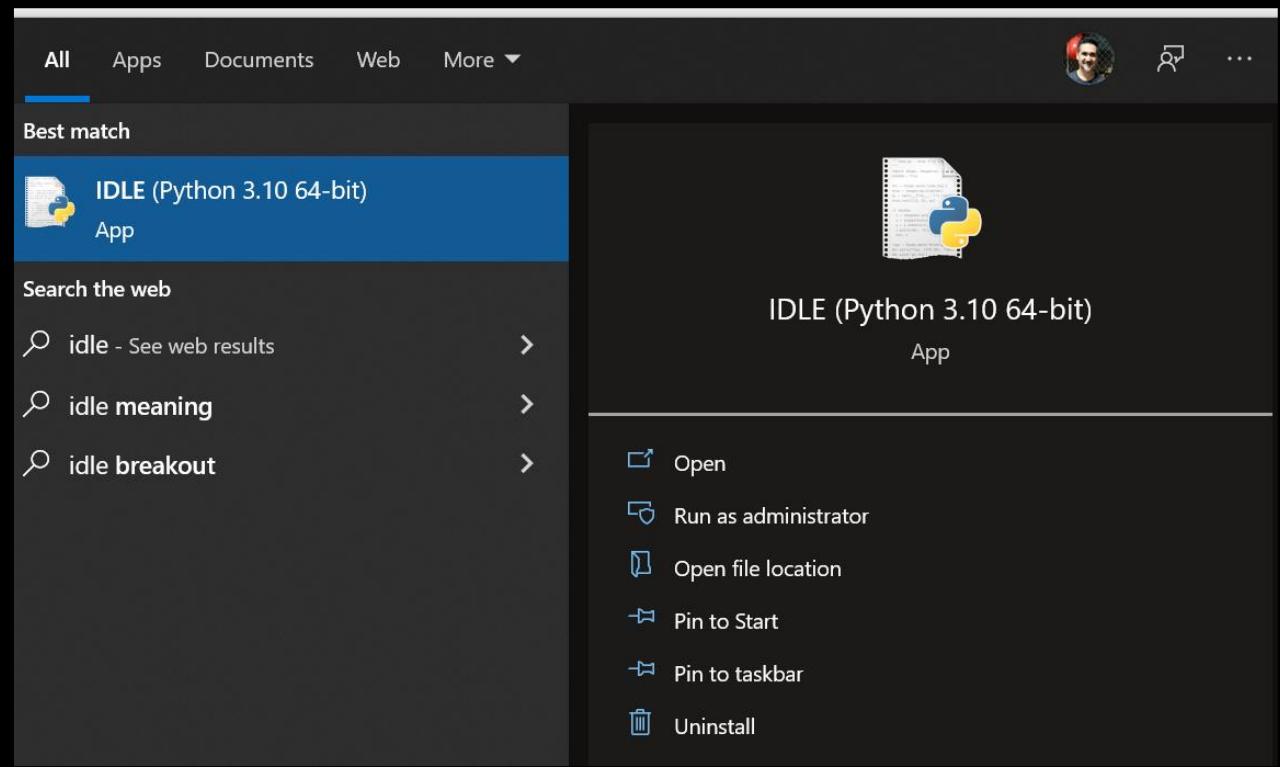


IMAGEM 40 –IDLE

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

Ambiente de desenvolvimento IDLE

- IDLE é um ambiente de desenvolvimento simplificado, modo texto, que possibilita a criação de dicionários de dados (data set), funções para utilizar com o dicionário e comandos para importar, executar e testar as funções e o próprio conjunto de dados.

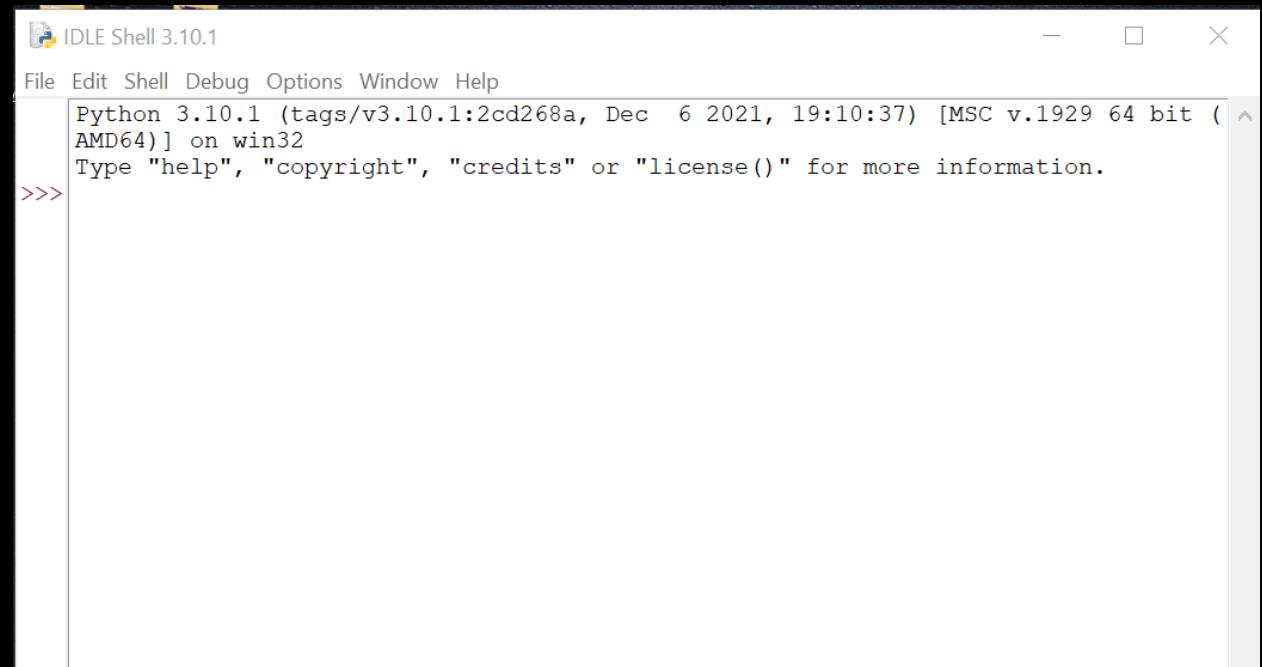


IMAGEM 41 –IDLE 2

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

Criando um Data set

- Crie um dicionário de dados chamado “clientes” que é composto por uma relação de nomes de clientes, uma lista de produtos contratados que esses clientes possuem em outra instituição e um valor de referência que está relacionado ao limite de crédito (no caso de cartão) e ao investimento (no caso dos outros produtos).

Salve o arquivo .py na pasta
\$Programs\Python\Python310\Lib



```
recomendacaoOF.py - C:\Users\jose.luiz.pagnossim\AppData\Local\Programs\Python\Python310\recomendacaoOF.py

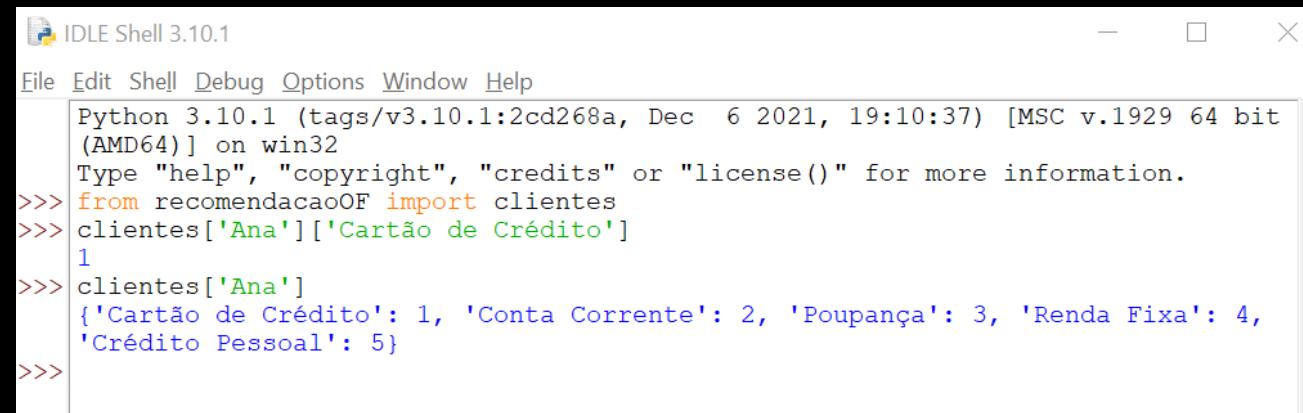
File Edit Format Run Options Window Help
clientes = {
    'Ana': {
        'Cartão de Crédito': 1,
        'Conta Corrente': 2,
        'Poupança': 3,
        'Renda Fixa': 4,
        'Crédito Pessoal': 5},
    'Marcos': {
        'Cartão de Crédito': 2,
        'Conta Corrente': 3,
        'Poupança': 4,
        'Renda Fixa': 5,
        'Renda Variável': 0.6},
    'Pedro': {
        'Cartão de Crédito': 3,
        'Conta Corrente': 4,
        'Poupança': 5,
        'Crédito Pessoal': 7},
    'Claudia': {
        'Cartão de Crédito': 4,
        'Conta Corrente': 5,
        'Poupança': 6}
}
```

Desenvolvendo um Sistema de Recomendação

Configuração do Ambiente Python (IDLE) e criação do Data set

Testando o Data set

- from recomendacaoOF import clientes
//importa a variável avaliações do dic
recomendacaoOF.py
- clientes['Ana']['Cartão de Crédito'] //Busca no cliente 'Ana' o valor da chave 'Cartão de Crédito'. Lembrando que é case sensitiv
- 1.0 //resultado apresentado
- Se quiser visualizar a lista completa dos produtos da 'Ana' digite apenas clientes['Ana']



The screenshot shows the Python 3.10.1 IDLE Shell interface. The window title is "IDLE Shell 3.10.1". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The shell area displays the following code and its output:

```
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> from recomendacaoOF import clientes
>>> clientes['Ana']['Cartão de Crédito']
1
>>> clientes['Ana']
{'Cartão de Crédito': 1, 'Conta Corrente': 2, 'Poupança': 3, 'Renda Fixa': 4,
 'Crédito Pessoal': 5}
```

IMAGEM 43 –DATA SET TESTE

Capítulo 5

5.2. Distância e Similaridade – Conceitos e cálculo

Distância e Similaridade – Conceitos e cálculo

ANTES DE INICIAR

Antes de calcular a distância vamos colocar nosso conjunto de dados em uma matriz para podermos visualizar a distância entre os clientes por meio de um gráfico de dispersão.

O GRÁFICO DE DISPERSÃO

O gráfico de dispersão indica o quanto dispersos os itens estão, dessa forma a similaridade se dá conforme encontramos dois itens mais próximos (menos distantes).

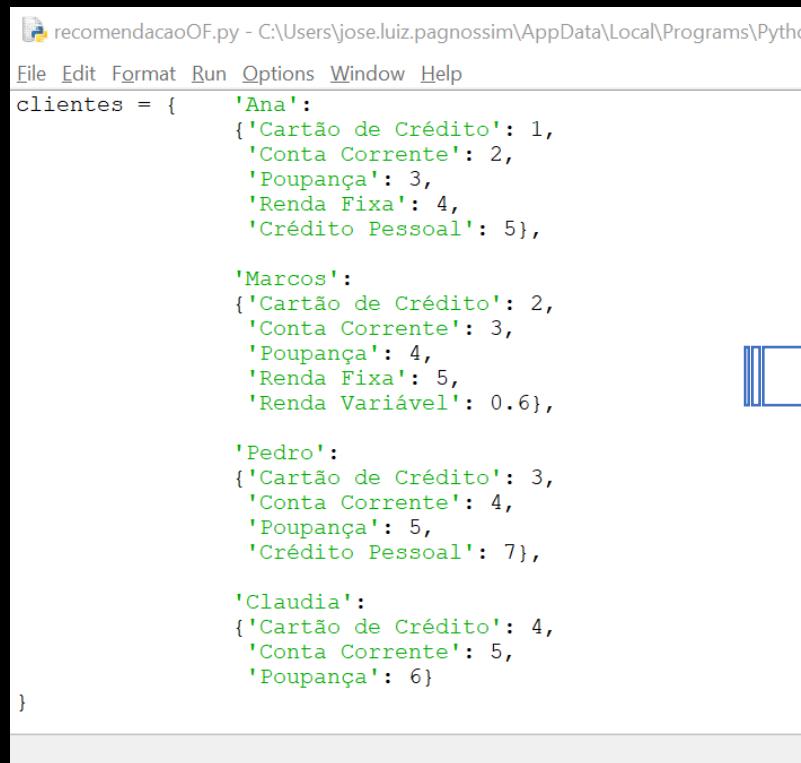
COORDENADAS

O Gráfico terá em um eixo um produto e no outro eixo outro produto, a coordenada {linha, coluna} representa a posição no gráfico que o cliente aparece de acordo com seus produtos e valores

Distância e Similaridade – Conceitos e cálculo

A Matriz completa representando o data set

- A matriz tem no eixo linha o cliente e no eixo coluna o produto, a coordenada {linha, coluna} representa a posição no gráfico que o cliente aparece de acordo com seus produtos e valores



```
recomendacaoOF.py - C:\Users\jose.luiz.pagnossim\AppData\Local\Programs\Python\clientes = {  
    'Ana':  
        {'Cartão de Crédito': 1,  
         'Conta Corrente': 2,  
         'Poupança': 3,  
         'Renda Fixa': 4,  
         'Crédito Pessoal': 5},  
  
    'Marcos':  
        {'Cartão de Crédito': 2,  
         'Conta Corrente': 3,  
         'Poupança': 4,  
         'Renda Fixa': 5,  
         'Renda Variável': 0.6},  
  
    'Pedro':  
        {'Cartão de Crédito': 3,  
         'Conta Corrente': 4,  
         'Poupança': 5,  
         'Crédito Pessoal': 7},  
  
    'Claudia':  
        {'Cartão de Crédito': 4,  
         'Conta Corrente': 5,  
         'Poupança': 6}  
}
```



Column1	Cartão de	Conta	Renda	Renda	Crédito	
	Crédito	Corrente	Poupança	Fixa	Variável	Pessoal
Ana	1,0	2,0	3,0	4,0		5,0
Marcos	2,0	3,0	4,0	5,0	6,0	
Pedro	3,0	4,0	5,0			7,0
Claudia	4,0	5,0	6,0			

IMAGEM 44 – MATRIZ DATA SET

Distância e Similaridade – Conceitos e cálculos

Representando os dados em um gráfico de dispersão

Passo a passo para criar o gráfico de dispersão (scatter) em Excel.

- Create blank Graphic scatter x y
- Select data
- Add legend entries
- Select series name and coordinates x, y
- Add data label
- Format data label, value from cell, select names,unselect x e y
- Vídeo de apoio para criação do gráfico
- <https://www.youtube.com/watch?v=IJKkr4M9ko>

	Conta	Corrente	Poupança
Column1	Corrente	Poupança	
Ana	2,0	3,0	
Marcos	3,0	4,0	
Pedro	4,0	5,0	
Claudia	5,0	6,0	

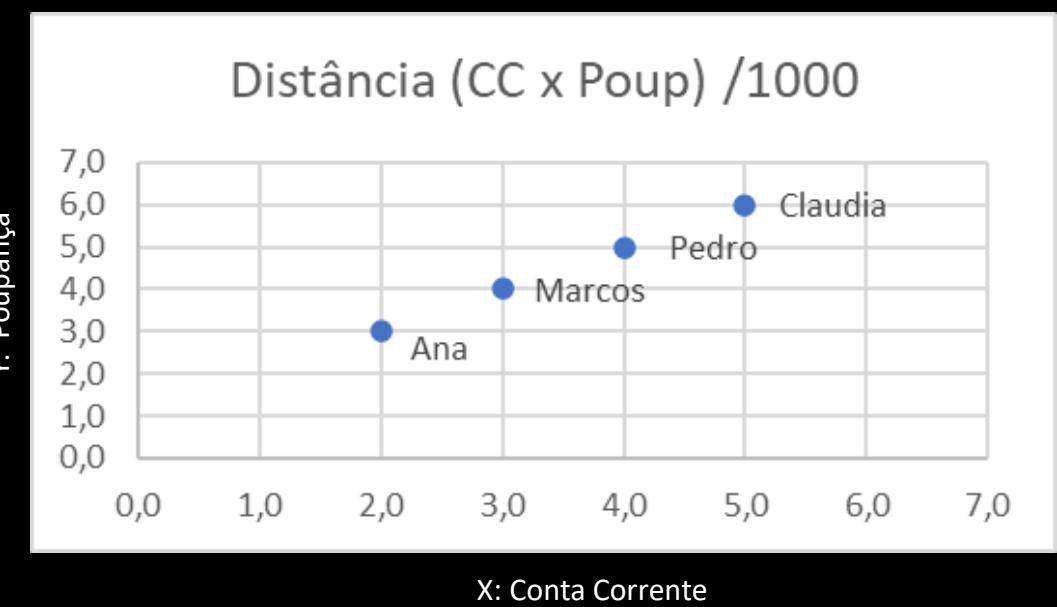
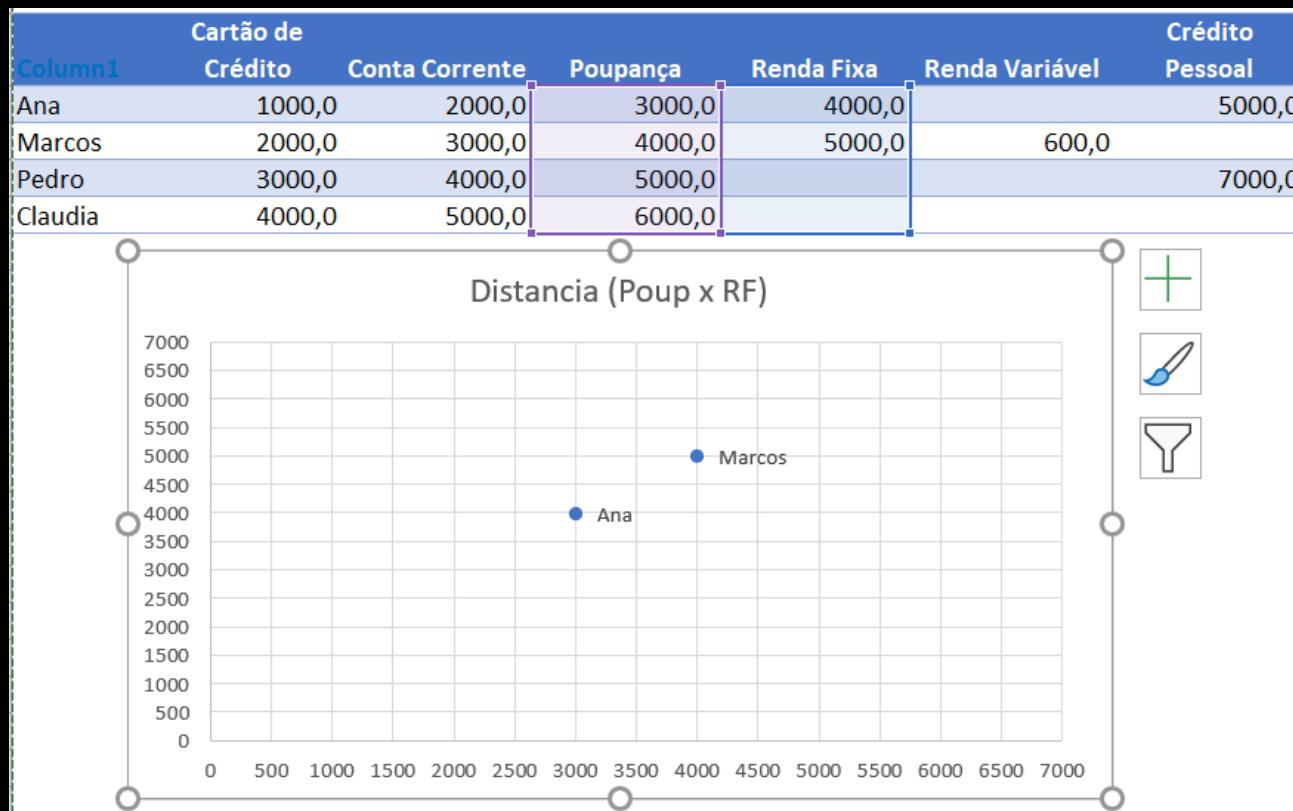


IMAGEM 45 – Gráfico 1

Distância e Similaridade – Conceitos e cálculos

Representando os dados em um gráfico de dispersão

Importante saber que o gráfico de dispersão para poder realizar o cálculo, leva em consideração somente clientes que tenham produtos em comum, caso contrário não é possível estabelecer as coordenadas x e y.



Obs. Note a importância de reduzir a escala para efeito de análise de dados, as coordenadas ficam distantes dificultando a análise visual dos dados.

IMAGEM 46 – Gráfico 2

Distância e Similaridade – Conceitos e cálculos

Fórmula da Distância Euclidiana e da Similaridade

Para Distância, quanto maior o indicador, menos similar; Para Similaridade, quanto maior o indicador, mais similar.

	i = 1	i = 2
	Conta Corrente	Poupança
Ana i (y)	2,0	3,0
Marcos i (x)	3,0	4,0
Subtração	-1,0	-1,0
Elevar ao quadrado	1,00	1,00
Soma	2,00	
Raiz Quadrada (DE)	1,41	
Similaridade	0,41	

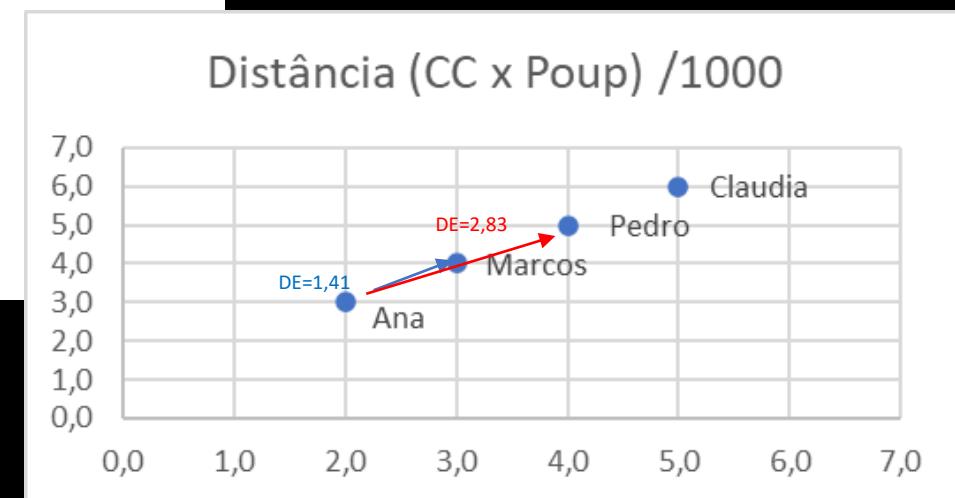
	Conta Corrente	Poupança
Ana i (y)	2,0	3,0
Pedro i (x)	4,0	5,0
Subtração	-2,0	-2,0
Elevar ao quadrado	4,00	4,00
Soma	8,00	
Raiz Quadrada (DE)	2,83	
Similaridade	0,26	

Distância Euclidiana (DE)

$$d(x, y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2}$$

Similaridade

$$\text{Sim} = 1 / (1 + \text{DE})$$



Capítulo 5

5.3. Implementando algoritmos em Python: distância,
similaridade e base para filtro colaborativo

Desenvolvendo um Sistema de Recomendação

Algoritmo para cálculo da distância euclidiana

```
from math import sqrt

def euclidiana(usuario1, usuario2):
    return sqrt(sum([pow(clientes[usuario1][item] - clientes[usuario2][item], 2)
                    for item in clientes[usuario1] if item in clientes[usuario2]]))
```

$$d(x, y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2}$$

IMAGEM 48 – Euclidiana 2

Verifica se o item do
usuario1 existe no usuario2

Obs. Implemente este algoritmo no mesmo arquivo do dicionário de dados para testar.

Desenvolvendo um Sistema de Recomendação

Algoritmo para cálculo da distância euclidiana

Testando o algoritmo de cálculo da função euclidiana

Salve o arquivo .py

Rode seu código: Run/Run Module (F5)

Volte ao IDLE

Importe a função euclidiana

Chame a função com diferentes parâmetros

The screenshot shows the Python IDLE interface. The top menu bar has 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. A context menu is open over some code, with 'Run Module' and 'F5' highlighted in blue. The main window displays a Python script and its execution in the IDLE Shell.

```
from recomendacaoOF import euclidiana
import math
from math import sqrt
def euclidiana(usuario1, usuario2):
    si = {}
    for item in usuarios[usuario1]:
        if item in usuarios[usuario2]:
            si[item] = 1
    if len(si) == 0:
        return 4.0
    else:
        d = 0
        for item in si:
            d += (users[item][usuario1] - users[item][usuario2]) ** 2
        return sqrt(d)
```

IDLE Shell 3.10.1

```
File Edit Shell Debug Options Window Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec 1 2021, 15:34:30) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information
>>> from recomendacaoOF import euclidiana
>>> euclidiana('Ana', 'Marcos')
2.0
>>> euclidiana('Ana', 'Pedro')
4.0
>>> euclidiana('Ana', 'Claudia')
5.196152422706632
>>>
```

Desenvolvendo um Sistema de Recomendação

Algoritmo para cálculo da similaridade

```
def similaridade(usuario1, usuario2):  
    de = euclidiana(usuario1, usuario2)  
    sim = 1 / (1+de) ←  
    return sim
```

Distância Euclidiana (DE)

$$d(x, y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2}$$

Similaridade

$$\text{Sim} = 1 / (1 + DE)$$

IMAGEM 49 - SIMILARIDADE

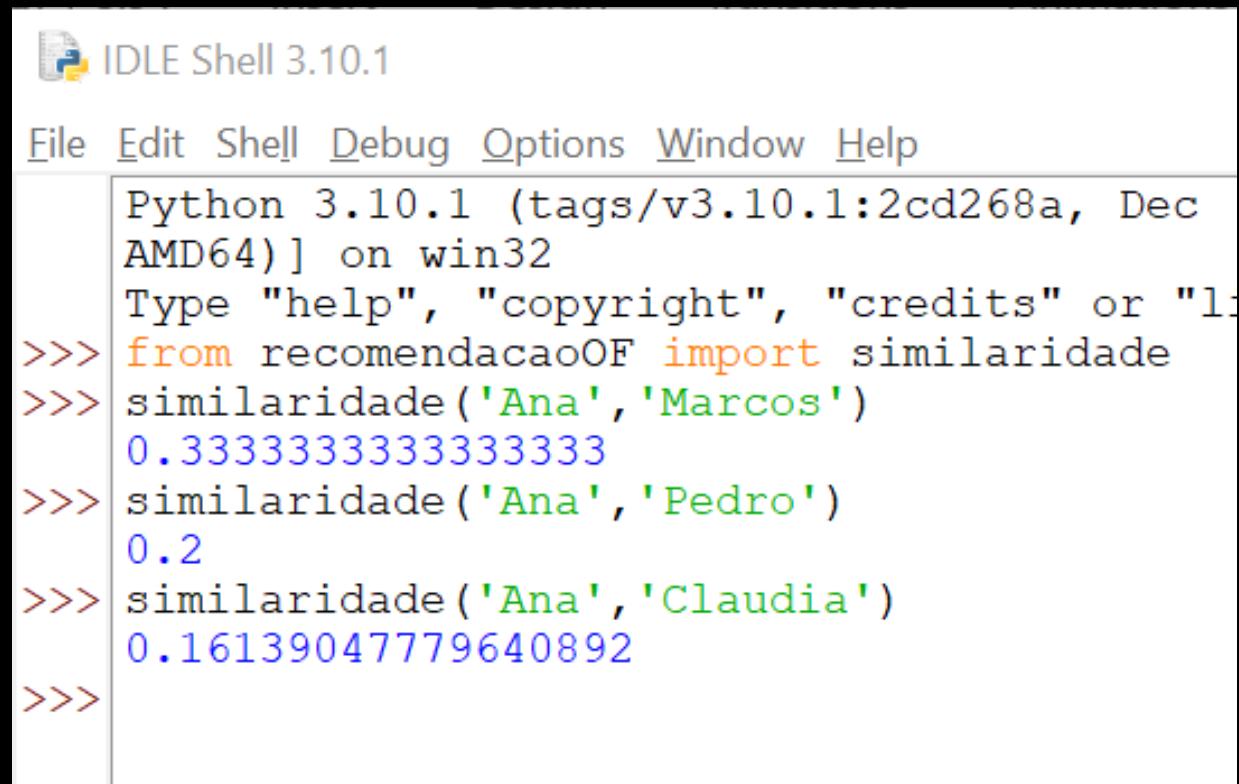
Obs. Implemente este algoritmo no mesmo arquivo do dicionário de dados para testar.

Desenvolvendo um Sistema de Recomendação

Algoritmo para cálculo da Similaridade

Testando o algoritmo de cálculo da função de similaridade

- Salve o arquivo .py
- Rode seu código: Run/Run Module (F5)
- Volte ao IDLE
- Importe a função similaridade
- Chame a função com diferentes parâmetros



The screenshot shows the Python IDLE Shell 3.10.1 interface. The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main window displays the following Python session:

```
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec AMD64) ] on win32
Type "help", "copyright", "credits" or "l:
>>> from recomendacaoOF import similaridade
>>> similaridade('Ana', 'Marcos')
0.3333333333333333
>>> similaridade('Ana', 'Pedro')
0.2
>>> similaridade('Ana', 'Claudia')
0.16139047779640892
>>>
```

IMAGEM 49 – TESTE SIMILARIDADE

Desenvolvendo um Sistema de Recomendação

Algoritmo para listar a similaridade entre usuários: base para filtro colaborativo

```
def getSimilares(usuario):
    listaSimilaridade = [(similaridade(usuario, outro), outro)
        for outro in clientes if outro != usuario]
    listaSimilaridade.sort()
    listaSimilaridade.reverse()
    return listaSimilaridade
```

1 Define a função que retorna a similaridade do usuário
2 Cria uma lista que chama a função similaridade passando
o usuário recebido via parâmetro e varrendo
a lista de clientes com outros usuários diferentes
3 Ordena a lista
4 Inverte a ordenação (para pegar a ordem decrescente)
5 retorna a lista de usuários similares

Obs. Implemente este algoritmo no mesmo arquivo do dicionário de dados para testar.

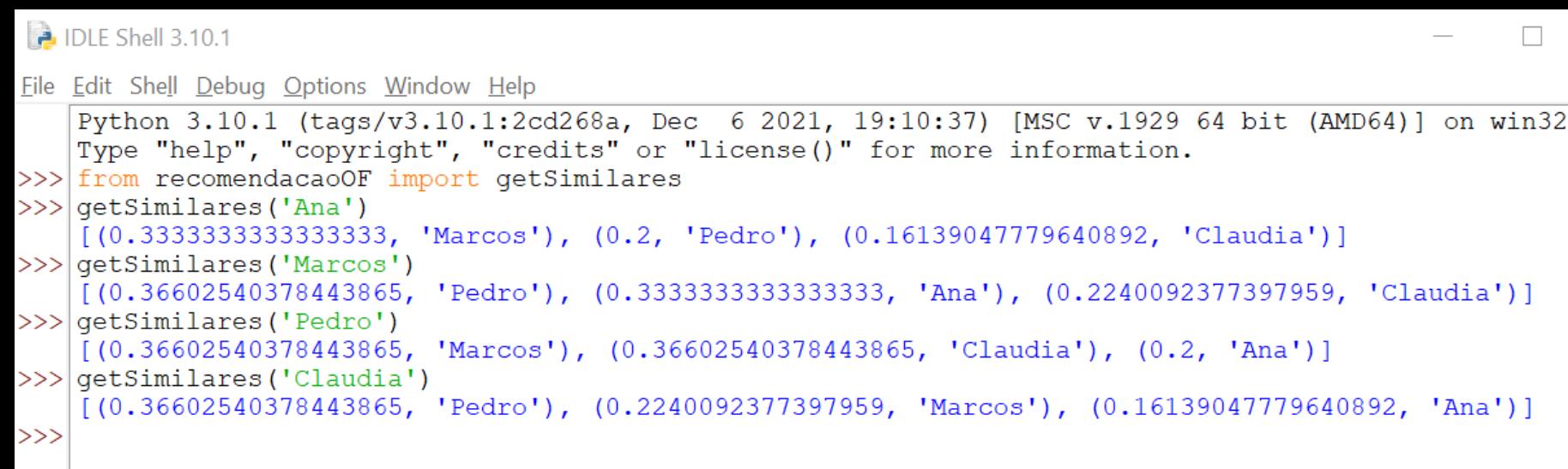
Desenvolvendo um Sistema de Recomendação

Algoritmo para retornar a lista de usuários similares

Testando o algoritmo que retorna a lista de usuários similares e o percentual de similaridade

- Salve o arquivo .py
- Rode seu código: Run/Run Module (F5). Volte ao IDLE
- Importe a função getSimilares
- Chame a função com diferentes parâmetros

O resultado desta
função é a base
para utilização da técnica
de filtro colaborativo



The screenshot shows the Python IDLE Shell 3.10.1 interface. The title bar says "IDLE Shell 3.10.1". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main window displays the following Python code and its output:

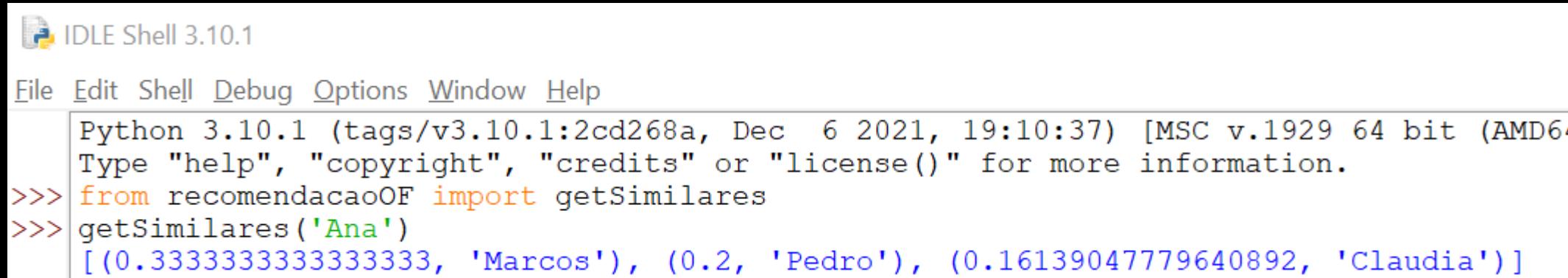
```
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> from recomendacaoOF import getSimilares
>>> getSimilares('Ana')
[(0.3333333333333333, 'Marcos'), (0.2, 'Pedro'), (0.16139047779640892, 'Claudia')]
>>> getSimilares('Marcos')
[(0.36602540378443865, 'Pedro'), (0.3333333333333333, 'Ana'), (0.2240092377397959, 'Claudia')]
>>> getSimilares('Pedro')
[(0.36602540378443865, 'Marcos'), (0.36602540378443865, 'Claudia'), (0.2, 'Ana')]
>>> getSimilares('Claudia')
[(0.36602540378443865, 'Pedro'), (0.2240092377397959, 'Marcos'), (0.16139047779640892, 'Ana')]
```

Desenvolvendo um Sistema de Recomendação

Conclusão

- Com o algoritmo de distância é possível determinar o quanto distantes estão dois usuários
- Com o algoritmo de similaridade é possível determinar o quanto similar são dois usuários
- Com o algoritmo que lista os usuários mais similares basta definir a melhor estratégia para o nosso sistema ou para o negócio e aplicar a técnica de filtro colaborativo. Por exemplo:
 - O algoritmo retorna que o usuário mais similar à Ana é o Marcos, dessa forma, seu sistema pode recomendar à Ana o último produto contratado pelo Marcos.



IDLE Shell 3.10.1

File Edit Shell Debug Options Window Help

```
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)]
Type "help", "copyright", "credits" or "license()" for more information.
>>> from recomendacaoOF import getSimilares
>>> getSimilares('Ana')
[(0.3333333333333333, 'Marcos'), (0.2, 'Pedro'), (0.16139047779640892, 'Claudia')]
```

Capítulo 5

5.4. Estudo de caso: Modelando e implementando uma prova de conceito de SR para Open Finance

Desenvolvendo um Sistema de Recomendação

Estudo de caso: Modelando e implementando uma prova de conceito de SR para Open Finance

Modelando uma base de dados em SQL-Server - Modelo a ser implementado no exemplo

- tbCliente: Tabela para cadastro dos clientes
- tbProduto: Tabela para cadastro dos produtos
- tbClienteProduto: Relacionamento entre Cliente e Produto com o valor que o cliente tem aplicado ou financiado no produto
- tbClienteSimilaridade: Auto-relacionamento com tbCliente para associar a similaridade entre clientes

```
clientes = {  
    'Ana':  
        {'Cartão de Crédito': 1,  
         'Conta Corrente': 2,  
         'Poupança': 3,  
         'Renda Fixa': 4,  
         'Crédito Pessoal': 5},  
  
    'Marcos':  
        {'Cartão de Crédito': 2,  
         'Conta Corrente': 3,  
         'Poupança': 4,  
         'Renda Fixa': 5,  
         'Renda Variável': 0.6},  
  
    'Pedro':  
        {'Cartão de Crédito': 3,  
         'Conta Corrente': 4,  
         'Poupança': 5,  
         'Crédito Pessoal': 7},  
  
    'Claudia':  
        {'Cartão de Crédito': 4,  
         'Conta Corrente': 5,  
         'Poupança': 6}  
}
```

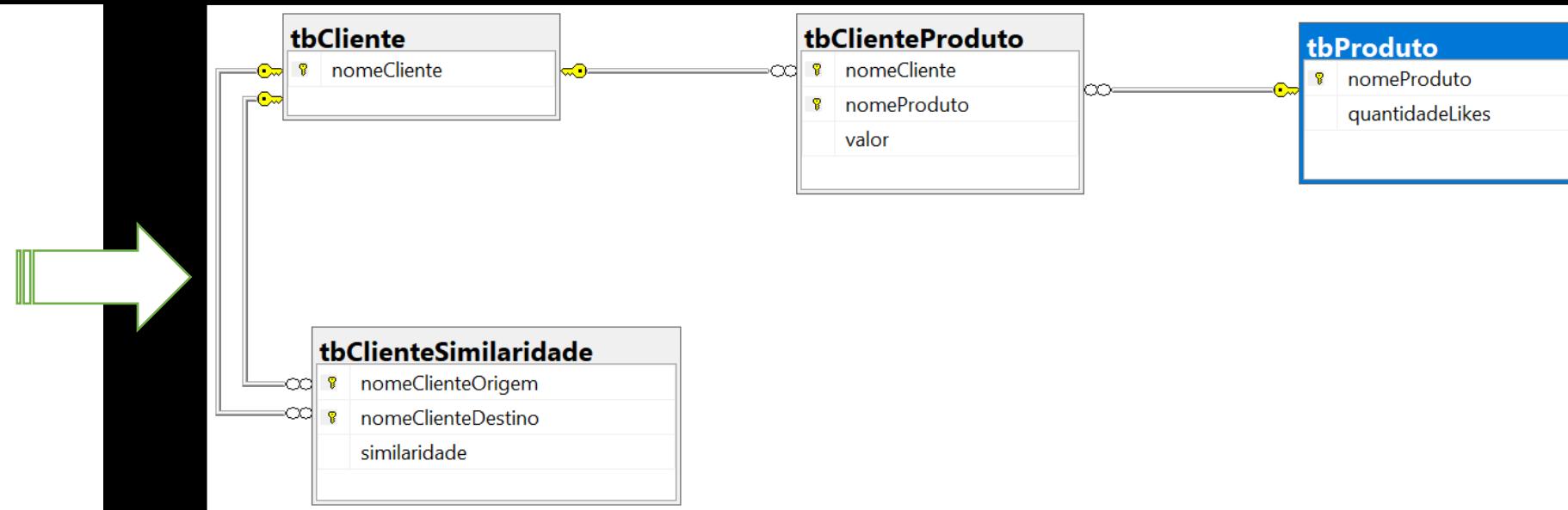


IMAGEM 52 – MODELO DE DADOS RELACIONAL

Desenvolvendo um Sistema de Recomendação

Estudo de caso: Modelando e implementando uma prova de conceito de SR para Open Finance

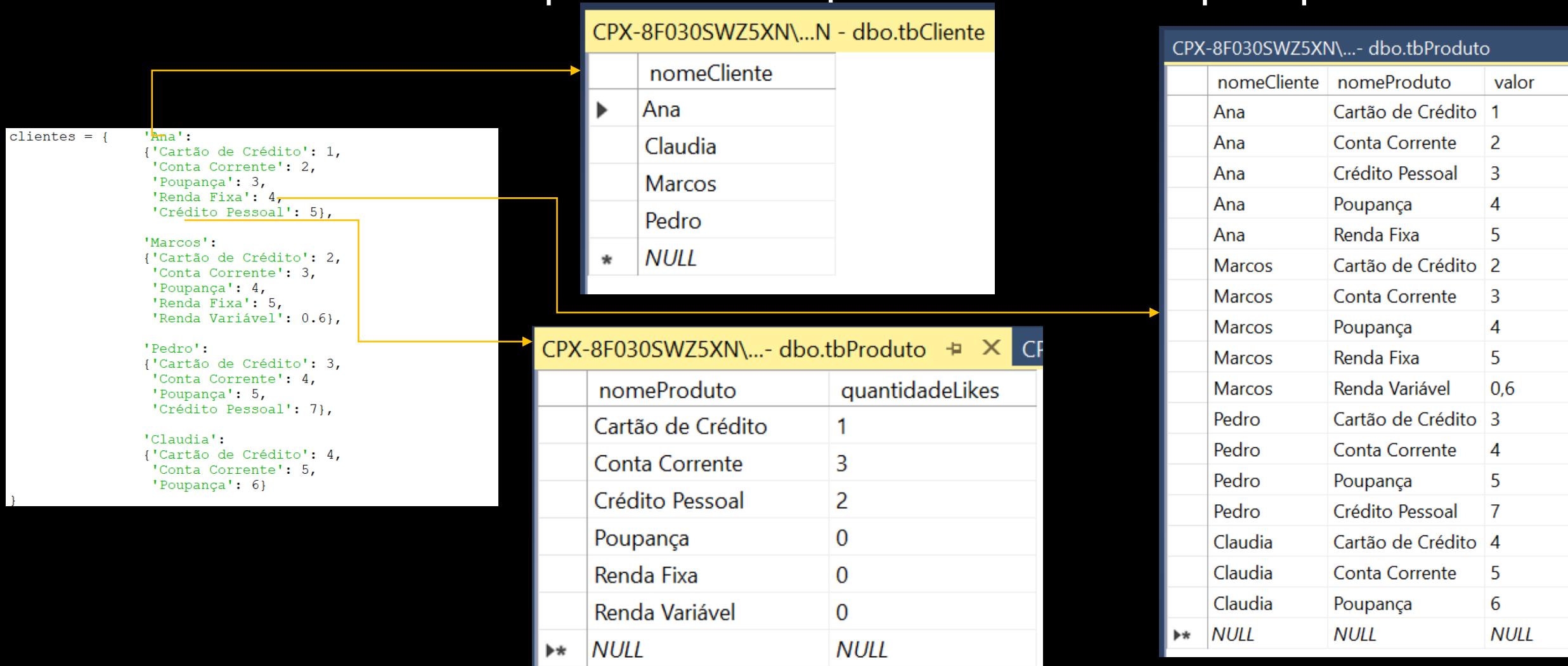


IMAGEM 53 – CARGA DOS DADOS NAS TABELAS

Desenvolvendo um Sistema de Recomendação

Carregando os resultados do algoritmo de similaridade do Python para a tabela na BD

The diagram illustrates the workflow for populating a database table with similarity data. On the left, a screenshot of the IDLE Shell 3.10.1 interface shows Python code running. A green arrow points from the shell output to the right, where a table titled "tbClienteSimilaridade" is displayed.

IDLE Shell 3.10.1

File Edit Shell Debug Options Window Help

Python 3.10.1 (tags/v3.10.1:2cd268a, Dec 6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

```
>>> from recomendacaoOF import getSimilares
>>> getSimilares('Ana')
[(0.3333333333333333, 'Marcos'), (0.2, 'Pedro'), (0.16139047779640892, 'Claudia')]
>>> getSimilares('Pedro')
[(0.36602540378443865, 'Marcos'), (0.36602540378443865, 'Claudia'), (0.2, 'Ana')]
>>> getSimilares('Marcos')
[(0.36602540378443865, 'Pedro'), (0.3333333333333333, 'Ana'), (0.2240092377397959, 'Claudia')]
>>> getSimilares('Claudia')
[(0.36602540378443865, 'Pedro'), (0.2240092377397959, 'Marcos'), (0.16139047779640892, 'Ana')]
```

tbClienteSimilaridade

	nomeClienteOrigem	nomeClienteDestino	similaridade
1	Ana	Claudia	0,16
2	Ana	Marcos	0,33
3	Ana	Pedro	0,2
4	Claudia	Ana	0,16
5	Claudia	Marcos	0,22
6	Claudia	Pedro	0,36
7	Marcos	Ana	0,33
8	Marcos	Claudia	0,22
9	Marcos	Pedro	0,36
10	Pedro	Ana	0,2
11	Pedro	Claudia	0,36
12	Pedro	Marcos	0,36

IMAGEM 54 – CARGA TABELA SIMILARIDADE

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

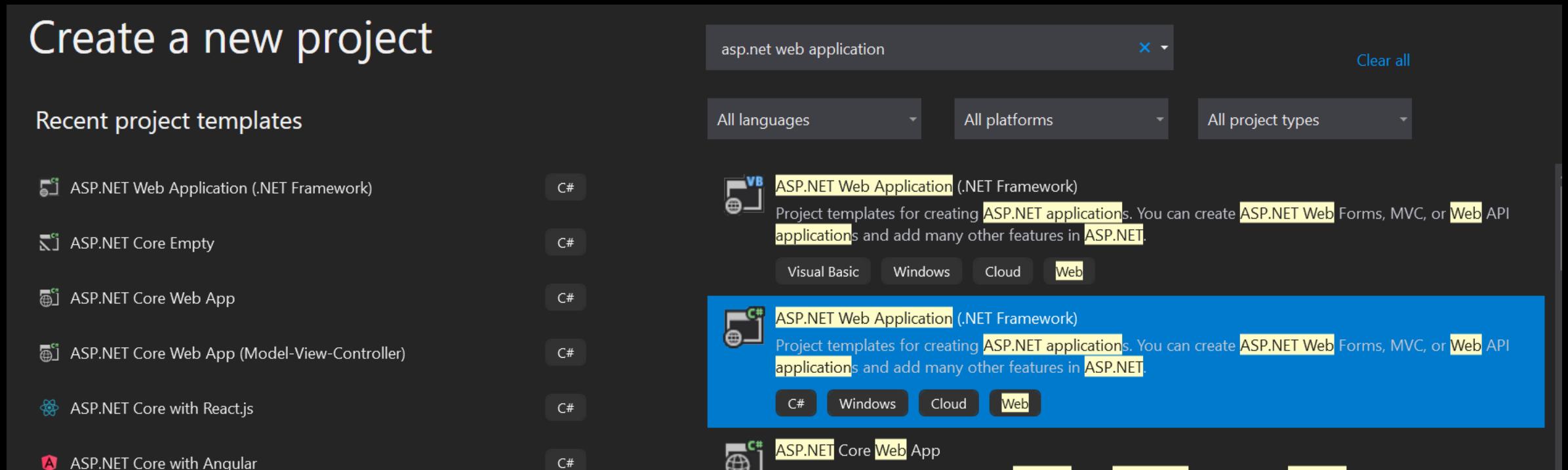


IMAGEM 55 – CRIANDO PROJETO WEB

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR
Defina um nome para sua aplicação e o local em que o projeto será salvo.

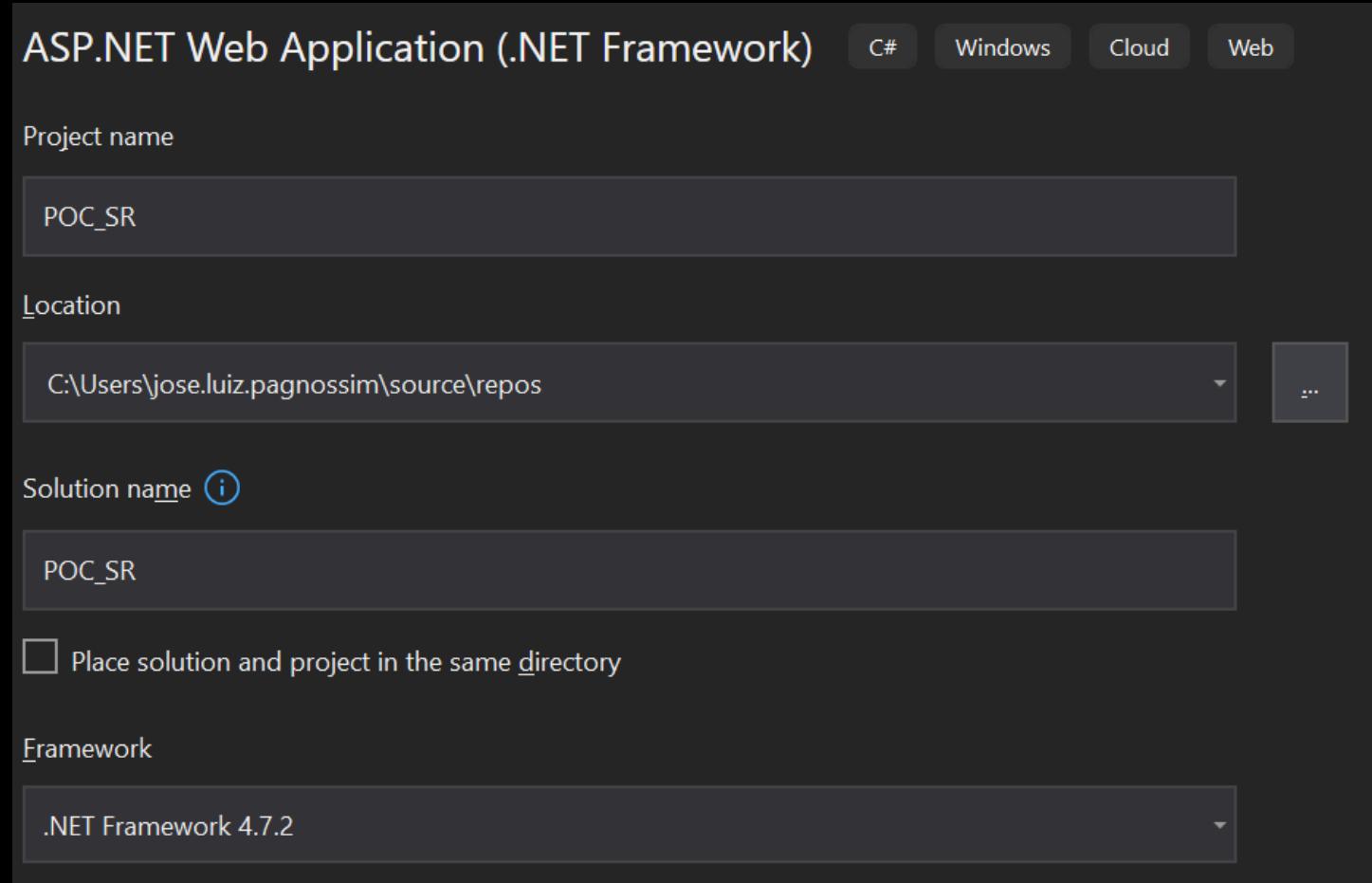
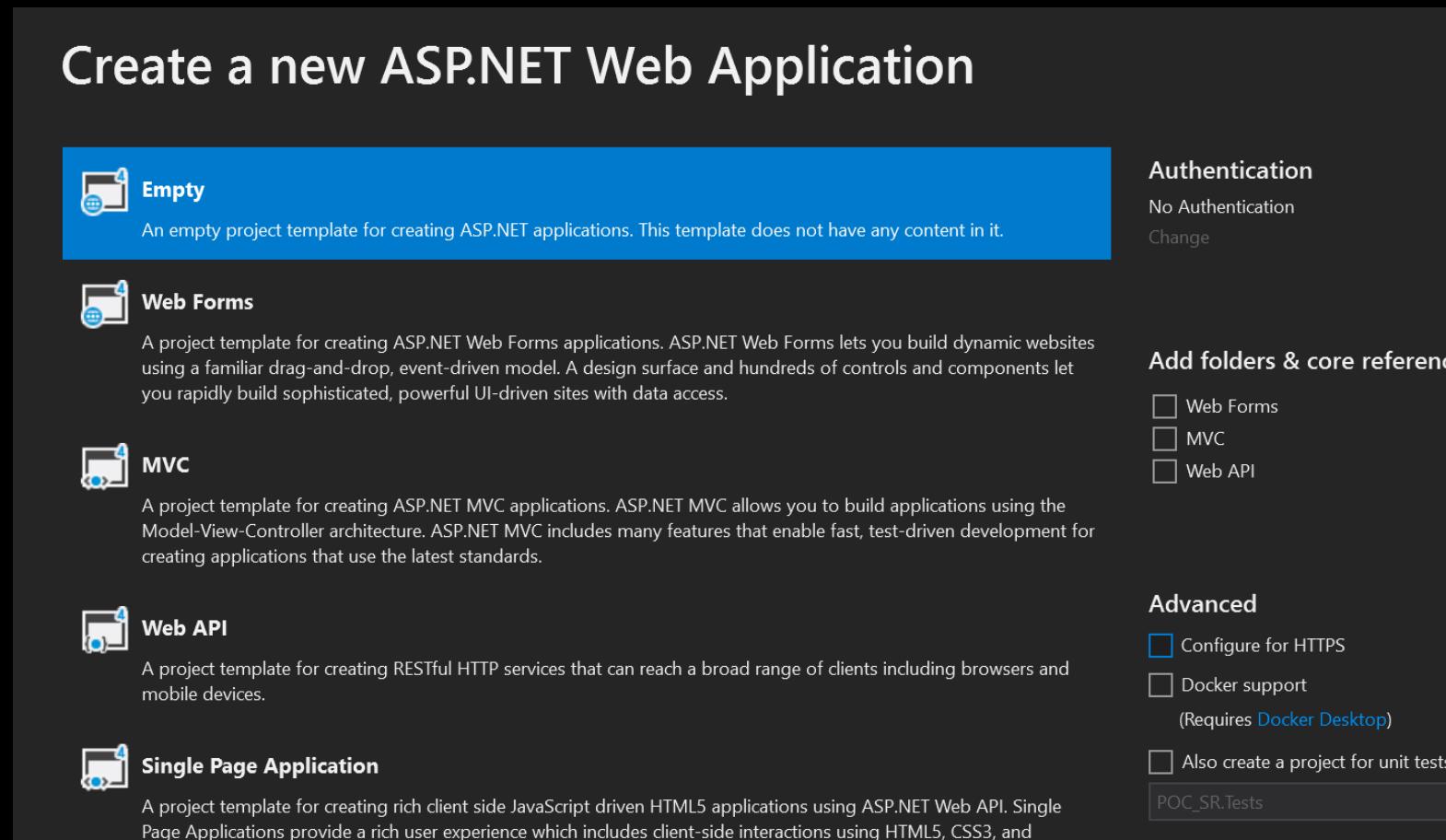


IMAGEM 56 –PROJETO WEB NOME

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR
Escolha a opção Empty e desmarque a opção Advanced/Configure HTTPS



Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR
Crie um novo webform para iniciar o desenvolvimento

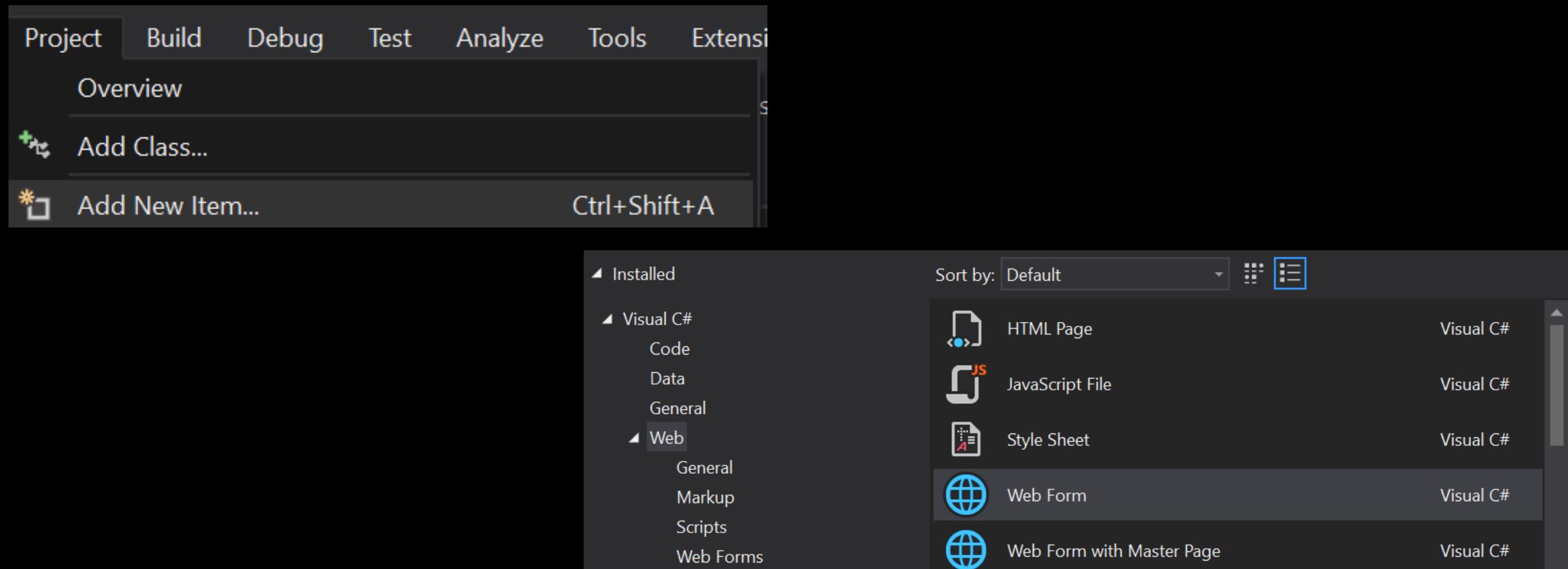


IMAGEM 58 – NOVO WEB FORM

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR
Defina o WebForm1.aspx como página de início e Na sequência, execute o projeto com IIS Express

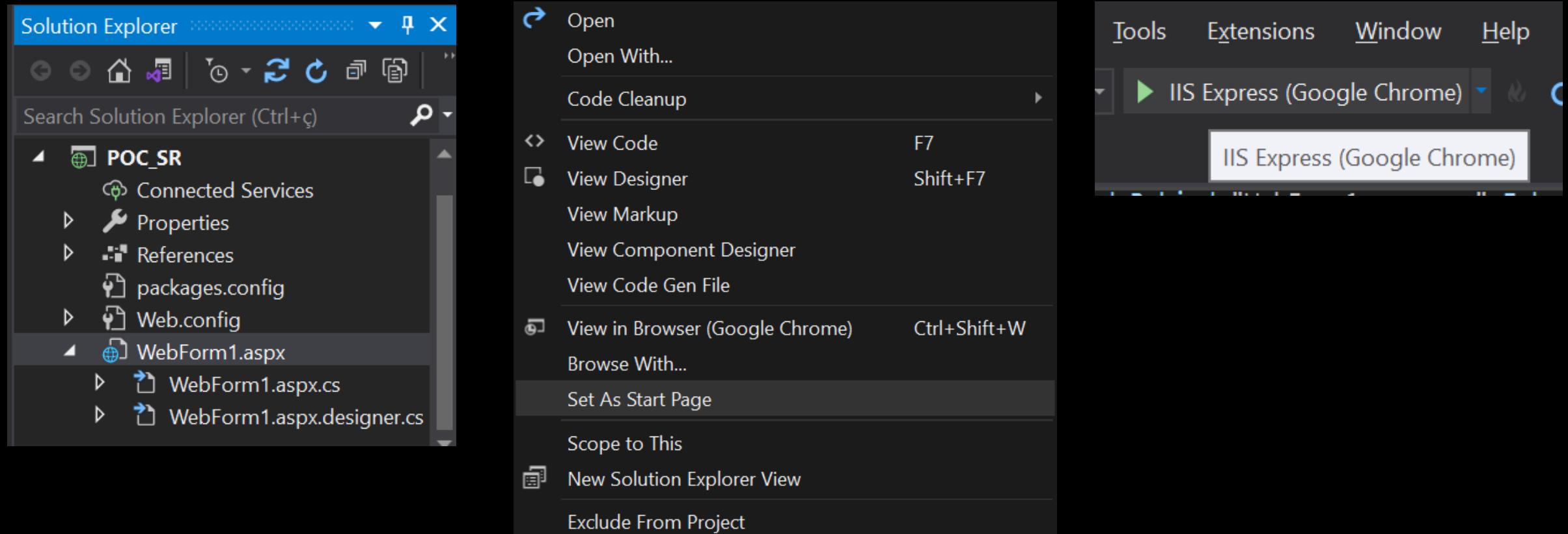


IMAGEM 59 – SET START PAGE

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Seu projeto será executado no navegador indicado na execução, neste caso no Google Chrome

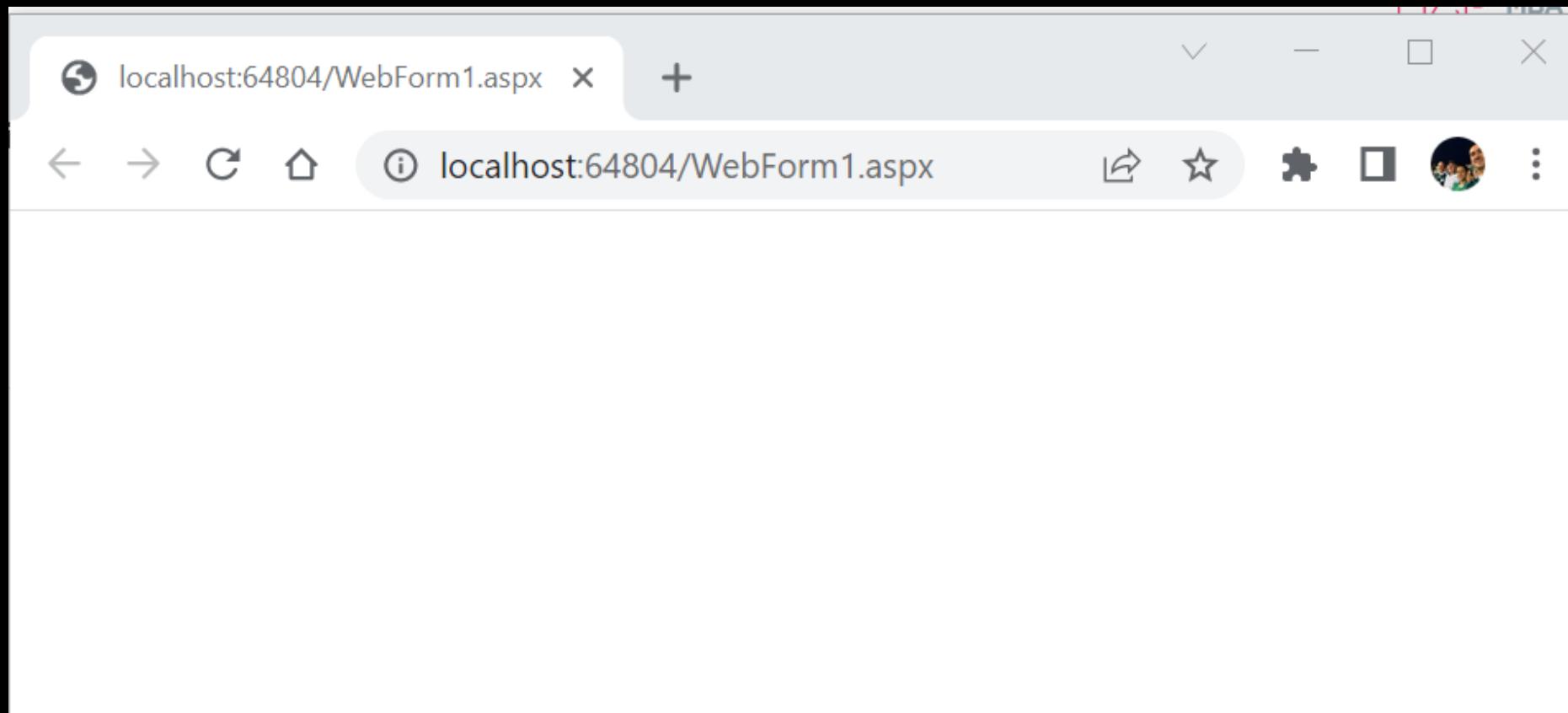


IMAGEM 60 – EXECUÇÃO PÁGINA WEB EM BRANCO

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR
A seguir o layout do frontend e as funcionalidades a serem implementadas.

Din Din Fintech #1 #3
Olá [Ana #2] . Ag. 1234 CC. 00012345-6 Atualizar Cliente #4

Recomendação Open Finance

nomeCliente	nomeProduto	valor
Marcos	Renda Variável	0,6

-

Recomendação Popularidade #5

nomeProduto	quantidadeLikes
Conta Corrente	3
Crédito Pessoal	2
Cartão de Crédito	2
Poupança	0
Renda Fixa	0
Renda Variável	0

-

Está curtindo os produtos da Din Din? #6 #7

 Cartão de Crédito	 Conta Corrente	 Crédito Pessoal
 Poupança	 Renda Fixa	 Renda Variável

[Resultado] #8

#1 - Cabeçalho da aplicação (fixo)
#2 - Dropdown com a listagem de todos os clientes
#3 - Agência e Conta (fixos)
#4 - Botão para atualizar os dados de recomendação, após seleção de um outro cliente no Dropdown
#5 - GridView com o resultado da recomendação por filtro colaborativo (Encontrar o cliente mais similar em relação ao cliente selecionado no DropDownList e selecionar os produtos que este cliente de destino possua e que o cliente de origem não possua)
#6 - GridView com o resultado da listagem dos produtos da tabela de produtos em ordem decrescente de quantidade de Likes
#7 - Espaço de colaboração que o usuário pode curtir ou descurtir um produto. Ao curtir, incrementar o campo quantidadeLikes da tabela de produto, ao descurtir, este campo deverá ser decrementado.
#8 - Label de resultado para apresentar mensagens de exceção geradas pela aplicação

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

No evento Load (carga do web form) são executados 4 blocos, o primeiro para preencher o dropdown com os nomes dos clientes, o segundo busca o usuário mais similar com base no cliente selecionado no dropdown, o terceiro popula o grid de recomendação com base na técnica de filtro colaborativo e por fim, o último grid carrega os produtos em ordem de maior popularidade. Esta é uma camada que integra o frontend ao backend que por enquanto está abstraído.

```
using System.Data;
Dados objDados = new Dados();
protected void Page_Load(object sender, EventArgs e) {
if (Page.IsPostBack == false) {
    //dropdown cliente
    DataTable dt = objDados.listarClientes();
    DropDownList1.DataSource = dt;
    DropDownList1.DataTextField = "nomeCliente";
    DropDownList1.DataBind();
    //usuário mais similar
    string strClienteMaisSimilar =
    objDados.consultarClienteMaisSimilar(DropDownList1.Text);
    //gridview recomendacao
    dt = objDados.listarRecomendacaoOF(DropDownList1.Text, strClienteMaisSimilar);
    GridView1.DataSource = dt;
    GridView1.DataBind();
    //gridview popularidade
    dt = objDados.listarProdutosPopularidade();
    GridView2.DataSource = dt;
    GridView2.DataBind(); } }
```

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Os botões de Curtiu/Descurtiu trabalham com uma lógica relacionada à imagem, dessa forma a imagem padrão indica não curtiu, ao clicar no botão o evento verifica se a imagem é de Descurtiu, se for, muda para a imagem de Curtiu e contabiliza o indicador (incrementa quantidadeLikes), o inverso funciona de forma análoga, decrementando o indicador de Likes e voltando a imagem para Descurtiu. O exemplo abaixo funciona para o botão de Cartão de Crédito, para os demais botões, basta usar o mesmo código alterando apenas o nome do componente de imagem e o produto que deseja incrementar ou decrementar.

```
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    if (ImageButton1.ImageUrl == "~/10_likeNao.png")
    {
        ImageButton1.ImageUrl = "~/10_likeSim.png";
        objDados.IncrementarLike("Cartão de Crédito");
    }
    else
    {
        ImageButton1.ImageUrl = "~/10_likeNao.png";
        objDados.DecrementarLike("Cartão de Crédito");
    }
}
```

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Com a parte do Frontend implementada e as chamadas aos métodos do objeto objDados que estávamos abstraindo até o momento, agora podemos desenvolver a camada de Backend, responsável pelas consultas e atualização dos dados na Base de Dados. Para resolver o erro abaixo, precisamos criar a classe Dados.



```
namespace POC_SR
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        Dados objDados = new Dados();
```

Os métodos de acesso a dados, devem ser implementados em uma classe que vamos chamar de 'Dados'. Para criar uma nova classe com esse nome, faça o seguinte

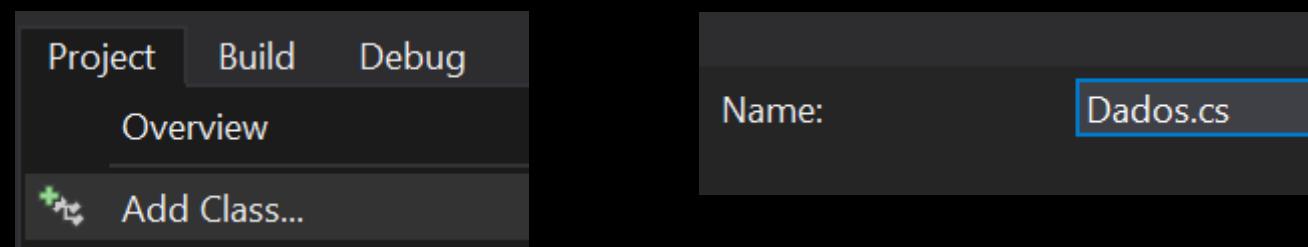
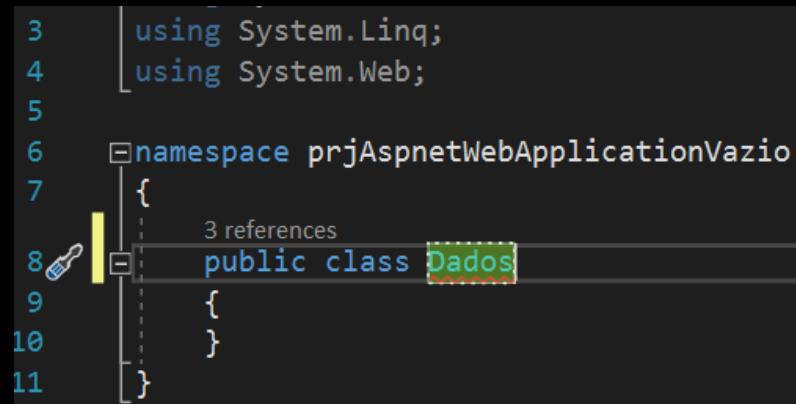


IMAGEM 62 – CLASSE DADOS

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Note que foi criada uma classe vazia.



```
3  using System.Linq;
4  using System.Web;
5
6  namespace prjAspnetWebApplicationVazio
7  {
8      public class Dados
9      {
10  }
11 }
```

A screenshot of a code editor showing a C# file. The code defines a class named 'Dados' within a namespace 'prjAspnetWebApplicationVazio'. The class is currently empty, containing only a constructor. The code is numbered from 1 to 11. A yellow vertical bar is positioned to the left of the code, and a tooltip '3 references' is visible near the class name 'Dados'.

IMAGEM 63 – CLASSE DADOS VAZIA

Agora vamos declarar as bibliotecas para acesso aos métodos de Dados e do Client do SQL-Server.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Data;
using System.Data.SqlClient;
namespace prjAspnetWebApplicationVazio{
    public class Dados
    {
    }
}
```

The same code editor screenshot as Image 63, but with several lines highlighted in yellow. These highlighted lines are the declarations for the 'System.Data' and 'System.Data.SqlClient' namespaces, as well as the opening brace of the 'Dados' class definition. This visual cue indicates which parts of the code have been modified or are being discussed.

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Agora vamos implementar o método listarClientes que consulta todos os clientes da tabela tbClientes e retorna um objeto do tipo DataTable (dt) para ser populado no dropdown.

```
SqlConnection con; //objeto responsável pela conexão com a BD
SqlCommand cmd; //objeto que implementa o comando SQL e associa à conexão
public DataTable listarClientes(){
    conectar(); //Conecta na BD do SQL-Server
    con.Open(); //Abre a conexão com a BD
    string sql = "SELECT nomeCliente FROM tbCliente"; //Busca os clientes da tabela tbCliente
    SqlCommand adoCmd = new SqlCommand(sql, con); //Associa o comando SQL com a conexão
    SqlDataAdapter da = new SqlDataAdapter(adoCmd); //Cria um adaptador associado ao comando SQL, esse adaptador é necessário para preencher os dados retornados no comando SQL para um objeto DataTable
    DataTable dt = new DataTable(); //Cria um objeto DataTable para receber o resultado do comando SQL
    da.Fill(dt); //Preenche o DataTable utilizando o método da.Fill do objeto SqlDataAdapter
    return dt; //Retorna o objeto dt (DataTable preenchido) }
```

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. At the top, there are two tabs: 'CPX-8F030SWZ5XN\... - dbo.tbProduto' and 'CPX-8F030SWZ5XN..NDIN - Diagram_0*'. Below the tabs, a query window displays the following SQL code:

```
/*
***** Seleciona todos os clientes da tabela tbCliente *****/
SELECT nomeCliente FROM tbCliente
```

Below the code, there are two tabs: 'Results' and 'Messages'. The 'Results' tab is selected and shows a table with four rows of data:

	nomeCliente
1	Ana
2	Claudia
3	Marcos
4	Pedro

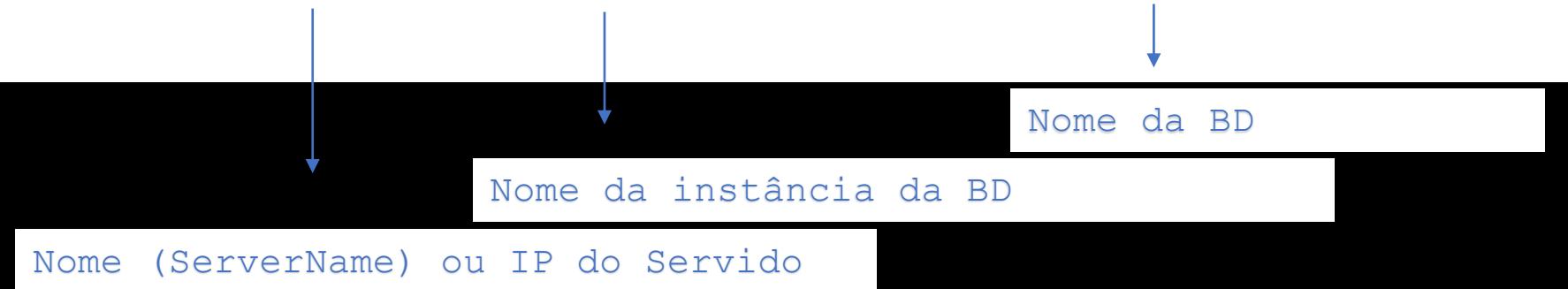
Este é o resultado que esperamos obter no retorno do banco de dados.

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

No método anterior, utilizamos um procedimento chamado conectar() responsável por definir os dados de conexão com a BD (string de conexão)

```
private void conectar()
{
    con = new SqlConnection("Data Source=CPX-8F030SWZ5XN\SQLEXPRESS01;Initial Catalog = DBDINDIN; Integrated Security = True");
}
```



Autenticação do Windows.
Há outras formas de se definir a string de conexão caso seja necessário especificar usuário e senha do SQL-Server. Exemplo:

```
"Persist Security Info=False;User ID=*****;Password=*****;Initial Catalog=AdventureWorks;Server=MySqlServer"
```

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Vamos ver agora como implementar a regra de Filtro Colaborativo, aproveitando o cálculo da Similaridade realizada em Python e carregada na nossa BD da aplicação. O primeiro passo é identificar qual é o cliente mais similar em relação ao usuário logado ou selecionado no nosso dropdown.

```
//Backend - classe Dados
public string consultarClienteMaisSimilar(string clienteOrigem) {
    conectar();
    con.Open();
    string sql = "select top 1 a.* from tbClienteSimilaridade a where a.nomeClienteOrigem = '" + clienteOrigem +
    "' order by similaridade desc;"; //Seleciona 1 registro da tabela que armazena a distância entre os
    clientes, passando como critério o cliente de origem (selecionado na tela), ordenado pela maior
    similaridade
    SqlCommand adoCmd = new SqlCommand(sql, con);
    SqlDataReader adoDR = adoCmd.ExecuteReader(); //Cria um objeto DataReader para armazenar o resultado da BD
    if (adoDR.HasRows){ //Se retornou algum registro
        if (adoDR.Read()) //Posiciona no registro retornado
            return adoDR["nomeClienteDestino"].ToString(); //Retorna o nome do cliente obtido
        else
            return "";
    }
    else
        return "";
}
```

	nomeClienteOrigem	nomeClienteDestino	similaridade
1	Ana	Marcos	0,33

Resultado
'Marcos' para a
cliente 'Ana'

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Agora que temos o nome do cliente com maior similaridade em relação ao cliente de origem, vamos fazer a recomendação do(s) produto(s) que o cliente de destino possui e o cliente de origem não possui (filtro colaborativo)

```
public DataTable listarRecomendacaoOF(string nomeClienteOrigem, string nomeClienteDestino){  
    conectar();  
    con.Open();  
    //Seleciona todos os campos da tabela tbClienteProduto onde o nomeCliente = nomeClienteDestino  
    // (mais similar) e o produto não exista para o cliente de origem  
    string sql = "select b.* from tbCliente a, tbClienteProduto b, tbProduto c ";  
    sql += "where a.nomeCliente = b.nomeCliente and b.nomeProduto = c.nomeProduto and b.nomeCliente =  
    '" + nomeClienteDestino + "' ";  
  
    sql += "and b.nomeProduto not in (select b.nomeProduto from tbCliente a, tbClienteProduto b,  
    tbProduto c ";  
    sql += "where a.nomeCliente = b.nomeCliente and b.nomeProduto = c.nomeProduto ";  
    sql += "and b.nomeCliente = '" + nomeClienteOrigem + "' )";  
    SqlCommand adoCmd = new SqlCommand(sql, con);  
    SqlDataAdapter da = new SqlDataAdapter(adoCmd);  
    DataTable dt = new DataTable();  
    da.Fill(dt);  
    return dt;  
}
```

	nomeProduto	valor
1	Renda Variável	600

Resultado para destino 'Marcos' e origem 'Ana'

IMAGEM 66 – RECOMENDAÇÃO FILTRO COLABORATIVO

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Neste método utilizamos o atributo `quantidadeLikes` como um indicador de popularidade e usamos a ordenação deste atributo de forma Decrescente para criar um ranqueamento por relevância com base na popularidade (curtidas)

```
//Backend
public DataTable listarProdutosPopularidade () {
conectar();
con.Open();
//Lista os produtos da tbProduto em ordem da maior para a menor popularidade
string sql = "select a.* from tbProduto a order by a.quantidadeLikes desc;";
SqlCommand adoCmd = new SqlCommand(sql, con);
SqlDataAdapter da = new SqlDataAdapter(adoCmd);
DataTable dt = new DataTable();
da.Fill(dt);
return dt;
}
```

	nomeProduto	quantidadeLikes
1	Conta Corrente	3
2	Crédito Pessoal	2
3	Cartão de Crédito	2
4	Poupança	0
5	Renda Fixa	0
6	Renda Variável	0

→ Resultado da listagem dos produtos por popularidade

IMAGEM 67 – LISTAR POPULARIDADE

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Métodos para incrementar ou decrementar o atributo `quantidadeLikes` com base na ação de like ou unlike do usuário pelos botões disponíveis para o usuário interagir com o sistema.

```
//Frontend - incrementa/decrementa Likes
if (ImageButton1.ImageUrl == "~/likeNao.png") {
    ImageButton1.ImageUrl = "~/likeSim.png";
    objDados.IncrementarLike("Cartão de Crédito");
} else {
    ImageButton1.ImageUrl = "~/likeNao.png";
    objDados.DecrementarLike("Cartão de Crédito");}
```

```
//Backend
public bool IncrementarLike(string produto) {
    conectar();
    cmd = new SqlCommand("UPDATE tbProduto SET
        quantidadeLikes = quantidadeLikes + 1 WHERE
        nomeProduto = '" + produto + "'", con);
    con.Open();
    cmd.ExecuteNonQuery();
    con.Close();
    return true;
}
```

```
//Backend
public bool DecrementarLike(string produto) {
    conectar();
    cmd = new SqlCommand("UPDATE tbProduto SET
        quantidadeLikes = quantidadeLikes - 1 WHERE
        nomeProduto = '" + produto + "'", con);
    con.Open();
    cmd.ExecuteNonQuery();
    con.Close();
    return true;
}
```

Desenvolvendo um Sistema de Recomendação

Criando uma aplicação Web para implementar e demonstrar as funcionalidades do SR

Executando o projeto e validando os resultados

The image shows a split-screen development environment. On the left, the Visual Studio IDE displays the 'POC_SR' project with the 'WebForm1.aspx.cs' file open. A red dot at line 40 indicates a break point has been set. The code implements logic for generating recommendations based on user input. On the right, a web browser window shows the application's output. It starts with a greeting message: 'Olá [Ana]'. Below it, two tables provide recommendations: 'Recomendação Open Finance' and 'Recomendação Popularidade'. The first table lists a single recommendation for 'Marcos' with 'Renda Variável' and a value of '0,6'. The second table lists five products with their respective like counts. At the bottom, a section asks if the user likes certain products, with thumbs-up icons next to each product name.

```
Process: [28840] iisexpress.exe
Any CPU | Continue | Debug | Lifecycle Events | Thread: [40020] Worker Thread | Stack
```

POC_SR

```
    32
    33     GridView2.DataBind();
    34 }
    35
    36
    37 protected void btnAtualizarCliente_Click(object sender, EventArgs e)
    38 {
    39     //gridview recomendacao
    40     string strClienteMaisSimilar = objDados.consultarClienteMaisSimilar(DropDown
    41     //gridview recomendação strClienteMaisSimilar | "Marcos" |
    42     DataTable dt = objDados.listarRecomendacaoOF(DropDownList1.Text, strClienteMaisSimila
    43     GridView1.DataSource = dt;
    44     GridView1.DataBind();
    45     //gridview popularidade
    46     dt = objDados.listarProdutosPopularidade();
    47     GridView2.DataSource = dt;
    48     GridView2.DataBind();
    49
    50
```

Din Din Fintech
Olá [Ana]. Ag. 1234 CC. 00012345-6 [Atualizar Cliente]

Recomendação Open Finance

nomeCliente	nomeProduto	valor
Marcos	Renda Variável	0,6

Recomendação Popularidade

nomeProduto	quantidadeLikes
Conta Corrente	4
Renda Variável	4
Crédito Pessoal	3
Cartão de Crédito	2
Poupança	0
Renda Fixa	0

Está curtindo os produtos da Din Din?

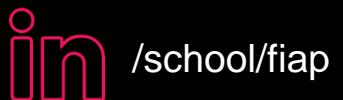
[Cartão de Crédito]	[Conta Corrente]	[Crédito Pessoal]
[Poupança]	[Renda Fixa]	[Renda Variável]

IMAGEM 69 – EXECUTANDO O PROJETO

MUITO OBRIGADO
E não deixe de realizar a survey
desta aula no portal FIAP.

 profjose.pagnossim@fiap.com.br

linkedin.com/in/jose-luiz-maturana-pagnossim-3454898



FIAP

Copyright © 2022 | Professor José Pagnossim

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente
proibido sem consentimento formal, por escrito, do professor/autor.

Capítulo 6

Avaliação, validação e tendências nos SRs

Capítulo 6

6.1. Avaliação dos SRs

Desenvolvendo um Sistema de Recomendação

Validação de um SR - ciclo de vida de um SR

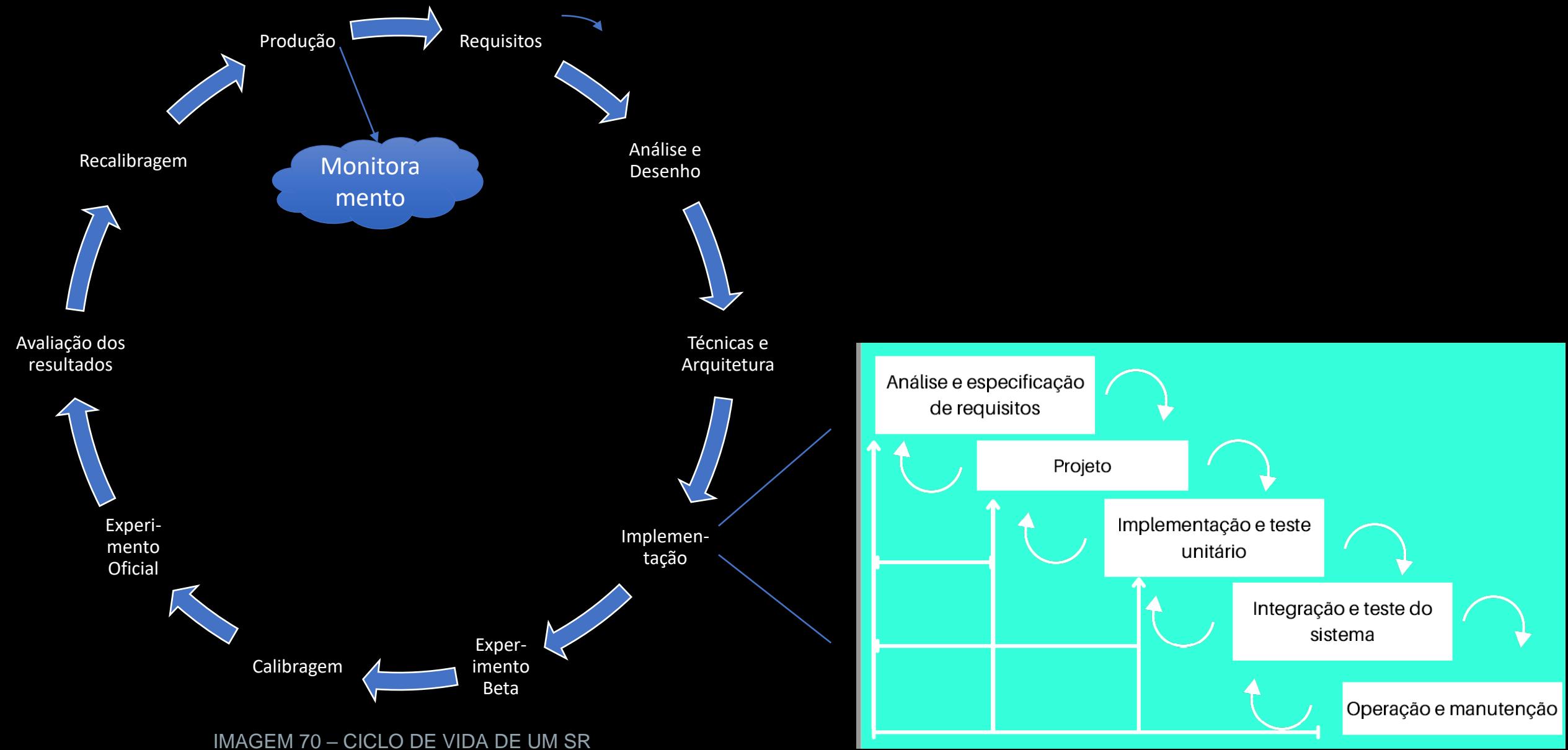
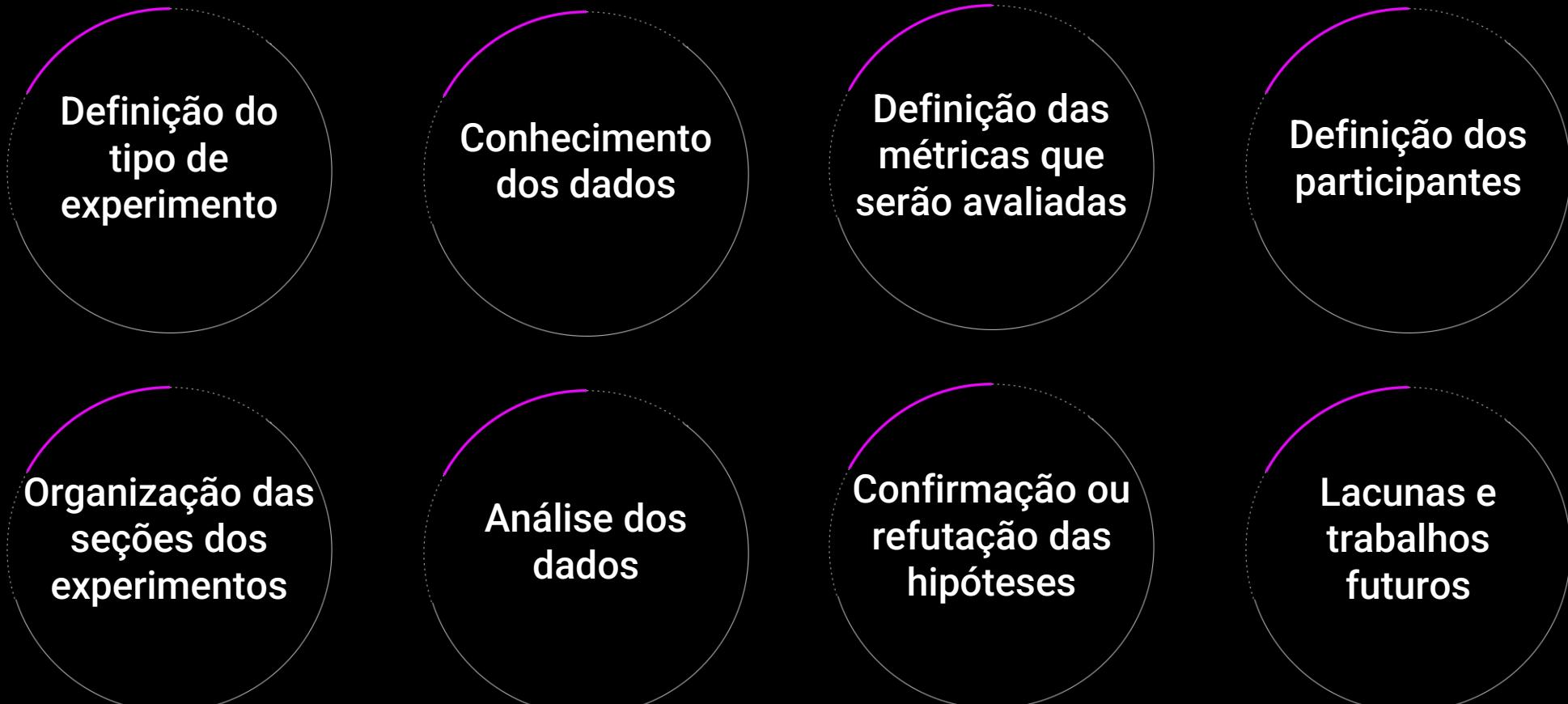


IMAGEM 70 – CICLO DE VIDA DE UM SR

Desenvolvendo um Sistema de Recomendação

Validação de um SR - ciclo de vida de um SR



Desenvolvendo um Sistema de Recomendação

Tipos de experimento

- O experimento **offline** é realizado usando um conjunto de dados pré-coletados de usuários selecionando ou classificando itens; o estudo do usuário, que é conduzido por meio de um conjunto de cenários de teste e estimulando os usuários a executarem tarefas de interação com o sistema, enquanto isso, seus comportamentos são observados e coletados.
- O método de avaliação **online** (Gunawardana2015) consiste em medir o comportamento do usuário quando este interage com diferentes sistemas de recomendação.
- Em um experimento **online**, utiliza-se alguma interface gráfica para interação com o usuário de forma a capturar os dados de navegação dos usuários durante a participação destes no experimento.
- Um experimento disponibilizado aos usuários reais por meio de uma experiência real de interface humano máquina que envolve a participação de seres humanos, requer aprovação pelo **comitê de ética** “CAAE 68557417.4.0000.5390”.

Desenvolvendo um Sistema de Recomendação

Conhecendo os Dados (Corpus)

Tabela 4 – Organização do *corpus* EBC de notícias

Canal da notícia	Total de notícias
Cidadania	96
Cultura	124
Educação	257
Esporte	137
Infantil	14
Notícias	344
Tecnologia	125

Fonte: José Luiz Maturana Pagnossim, 2018

Desenvolvendo um Sistema de Recomendação

Formas de avaliação (métricas)

Formas de avaliação

- {AV₁} Estudo da preferência do usuário
- {AV₂} Ranqueamento de item (implementado por meio da *MURR*)
- {AV₃} Cobertura do item no espaço
- {AV₄} Cobertura do usuário no espaço
- {AV₅} *Cold start* de item
- {AV₆} Novidade
- {AV₇} Diversidade
- {AV₈} Utilidade
- {AV₉} Serendipidade
- {AV₁₀} Similaridade de conteúdo
- {AV₁₁} Similaridade entre usuários
- {AV₁₂} Similaridade intragrupos
- {AV₁₃} Similaridade intergrupos
- {AV₁₄} Popularidade
- {AV₁₅} Estratégias de recomendação

Fonte: José Luiz Maturana Pagnossim, 2018

Desenvolvendo um Sistema de Recomendação

Definição dos participantes

- Grupo A - estudantes de graduação do curso de bacharelado de sistemas de informação da EACH/USP.
- Grupo B - estudantes de pós-graduação do curso de mestrado em sistemas de informação da EACH/USP.
- Grupo C - estudantes de graduação dos cursos de bacharelado em ciência da computação e tecnologia em análise e desenvolvimento de sistemas da Faculdade Carlos Drummond de Andrade (campus Tatuapé, São Paulo-SP).
- Grupo D - pessoas das áreas de tecnologia da informação e educação que fazem parte do convívio profissional do pesquisador.

As pessoas foram convidadas pelo pesquisador e pela orientadora desta pesquisa, por meio de uma mensagem eletrônica formal contendo os principais objetivos do experimento e endereço na internet para acesso ao sistema.

Desenvolvendo um Sistema de Recomendação

Organização das sessões do experimento

- Sessão 1 - testes do sistema de recomendação. Para testar o sistema foi selecionado um sub-grupo do grupo B especializado em sistemas de dados. Este sub-grupo contribuiu com a identificação de melhorias. Ao final desta sessão, uma nova versão do sistema foi criada com as correções dos defeitos encontrados. Os dados de participação dos participantes, foram descartados para efeito de apuração.
- Sessão 2 - estabelecimento de uma base inicial de navegação para comparação com as próximas sessões do experimento e análise dos efeitos de *cold start* de itens e *cold start* de usuários. Para esta sessão foi convidado um sub-grupo de pessoas selecionado a partir dos quatro grupos definidos na seção 4.2.1. Ao final do período estabelecido para essa sessão (uma semana), foram registradas trinta participações. Os dados foram separados para uso na sessão seguinte.
- Sessão 3 - comparação entre sessões em um cenário com dados já registrados pelo sistema de recomendação. Para esta sessão foi selecionado a partir dos quatro grupos definidos (os mesmos convidados anteriormente). Foi definido o período de participação (uma semana), mas após três dias da abertura de participação, o número de participações havia superado a sessão anterior. Neste momento, os dados para comparação com a sessão anterior foram encerrados e a partir deste momento foram contabilizadas novas participações (trinta e duas participações e, apesar do número de participações ter sido maior, foram registradas trinta e duas participações).
- Sessão 4 - nesta sessão participaram pessoas que acessaram o experimento após o encerramento da sessão 3, ou seja, eram pessoas que foram convidadas para participarem da sessão 2 ou da sessão 3, mas não tinham conseguido acessar o experimento dentro dos prazos estabelecidos nas respectivas sessões. Os dados desta sessão foram separados para serem comparados com as sessões 2 e 3. Apesar do número de participações ter sido maior nesta última sessão (quarenta e nove), foram gerados indicadores, tabelas e gráficos, a partir dos dados coletados, de forma a relativizar a análise dos resultados.

Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados)

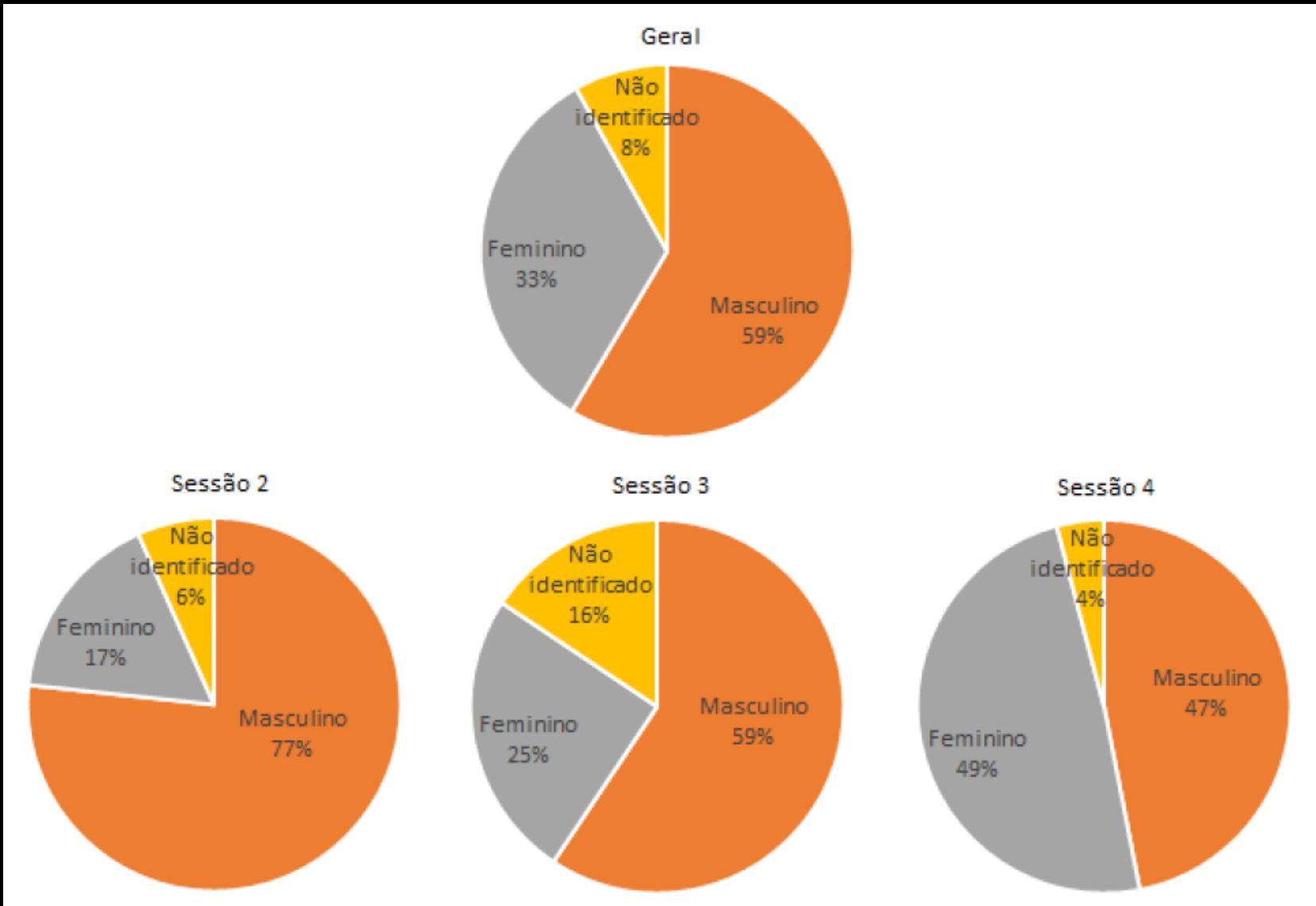


IMAGEM 74 – PARTICIPAÇÃO POR GÊNERO

Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados)

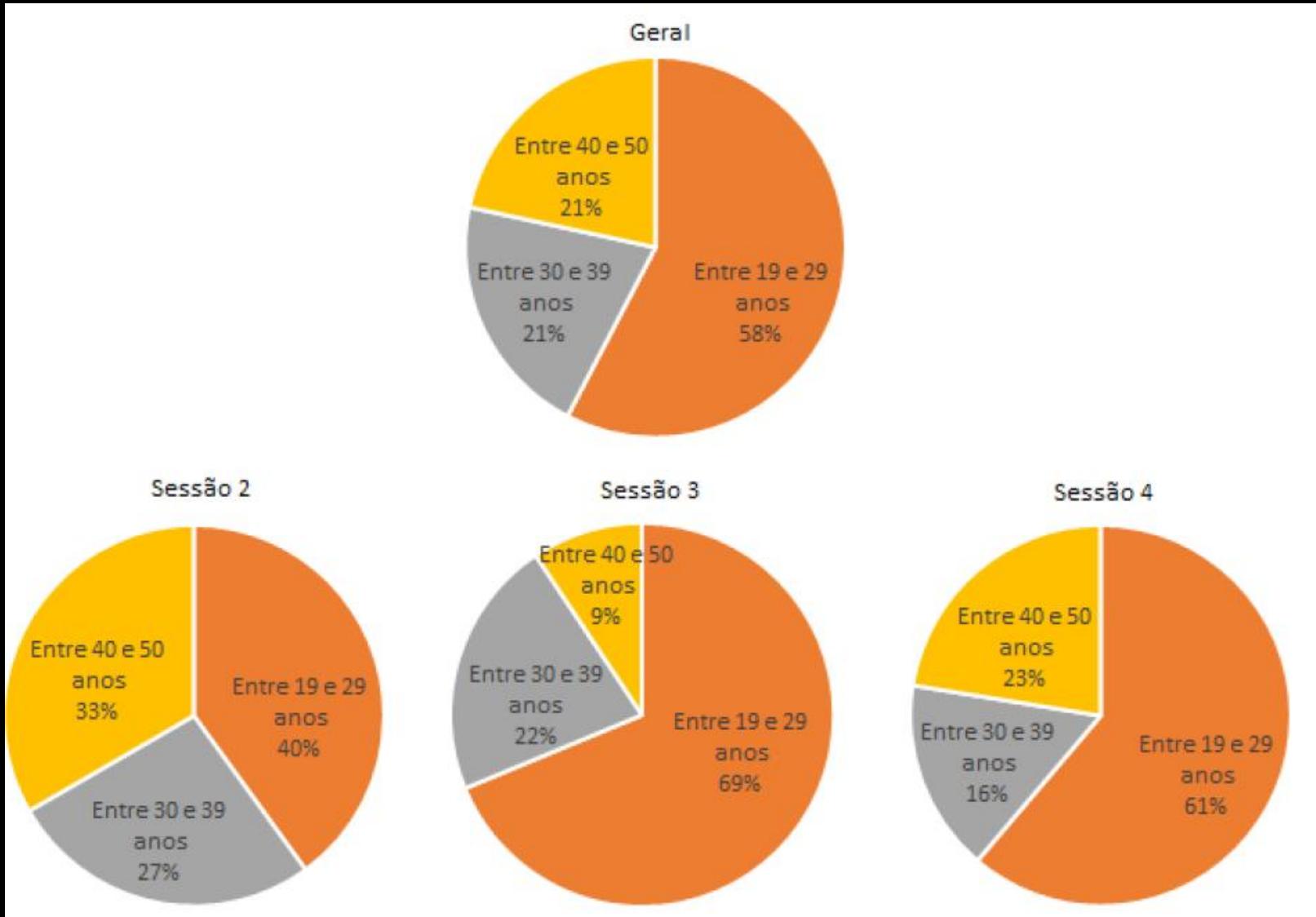


IMAGEM 75 – PARTICIPAÇÃO POR FAIXA ETÁRIA

Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados)

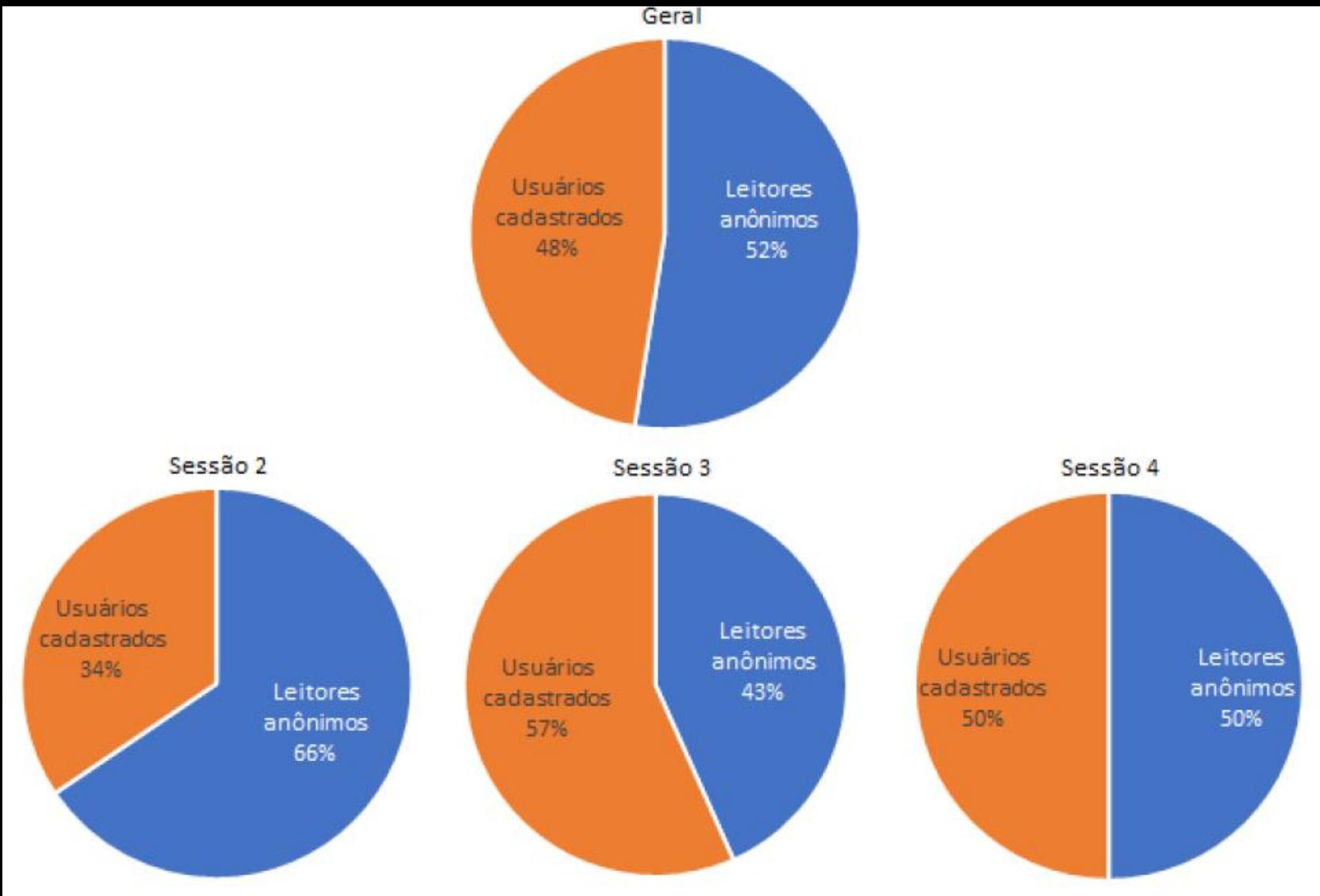


IMAGEM 74 – PARTICIPAÇÃO MODO DE NAVEGAÇÃO

Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados)

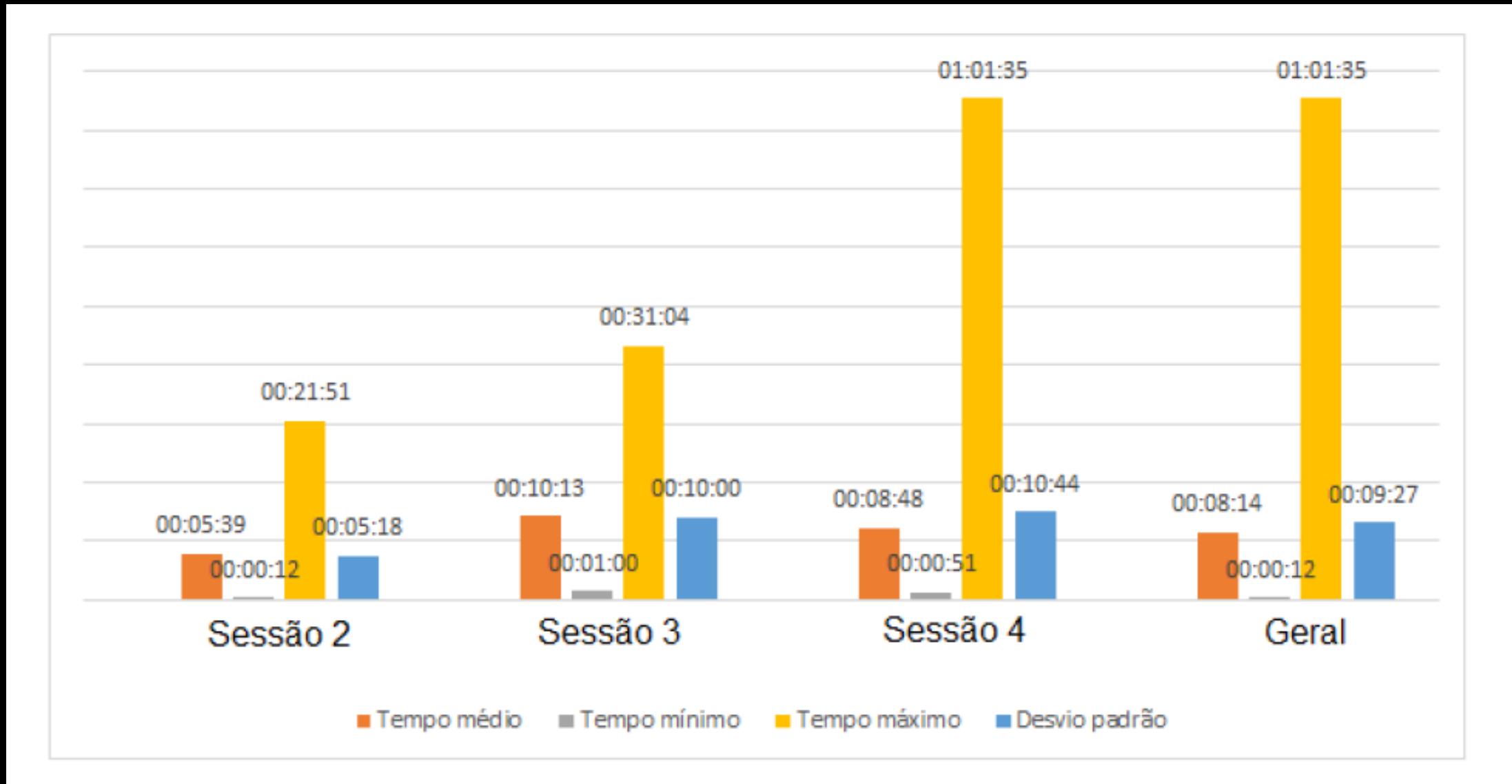


IMAGEM 75 – TEMPO DE PARTICIPAÇÃO POR SESSÃO

Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados) - verificação da eficácia

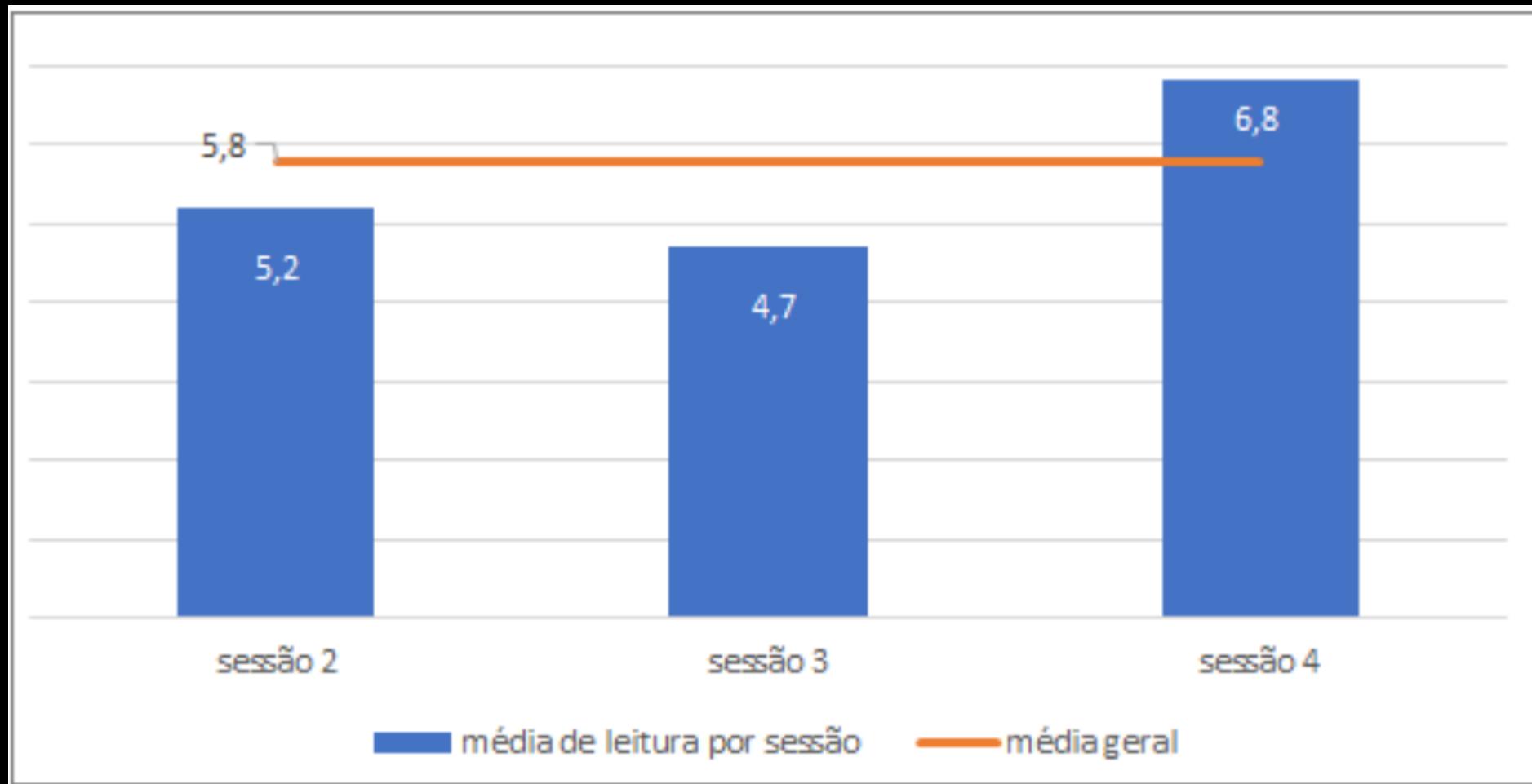


IMAGEM 76 – MÉDIA DE LEITURA POR SESSÃO

Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados) - verificação da eficácia

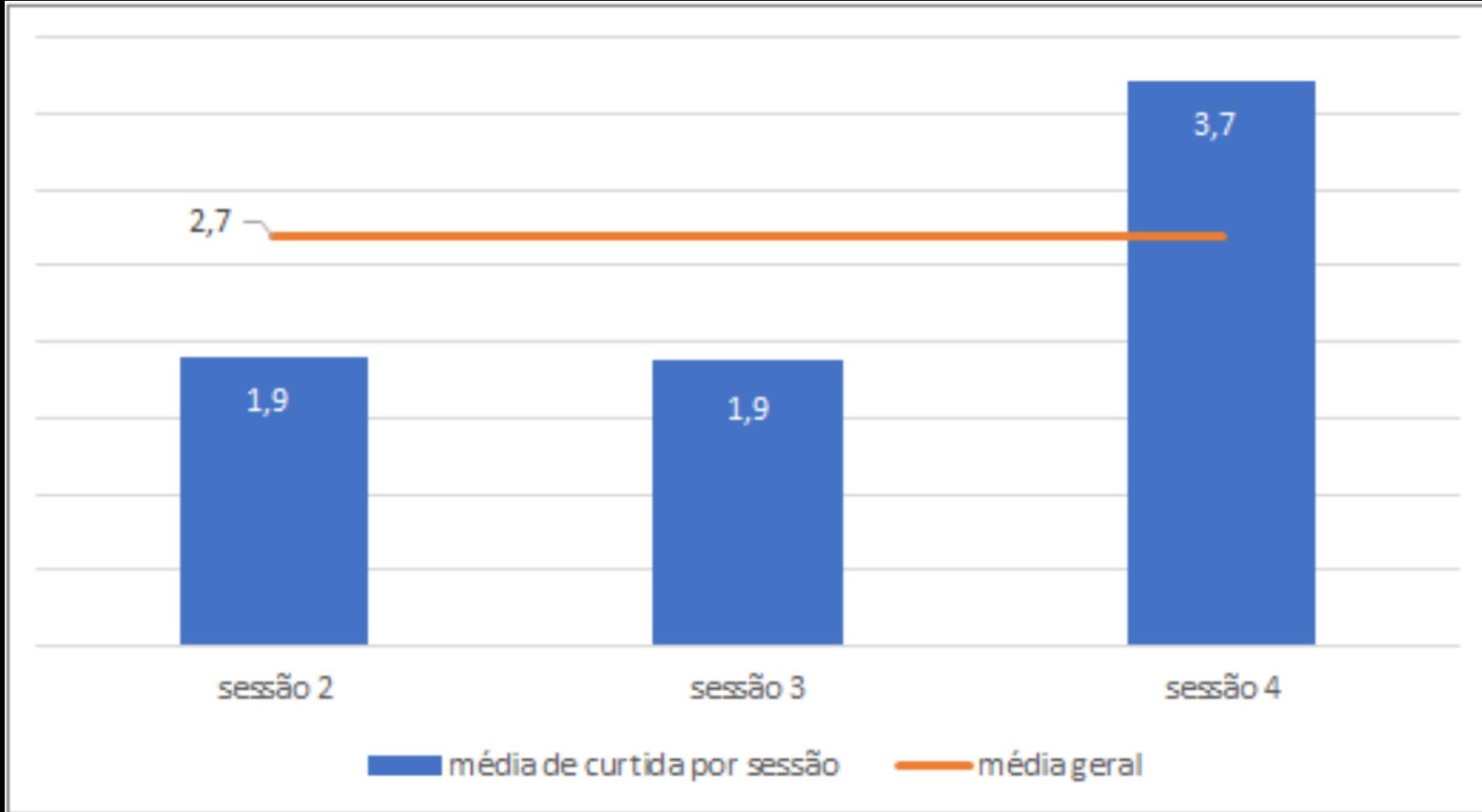


IMAGEM 77 – MÉDIA DE CURTIDA POR SESSÃO

Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados) - verificação da eficácia e confirmação da primeira hipótese

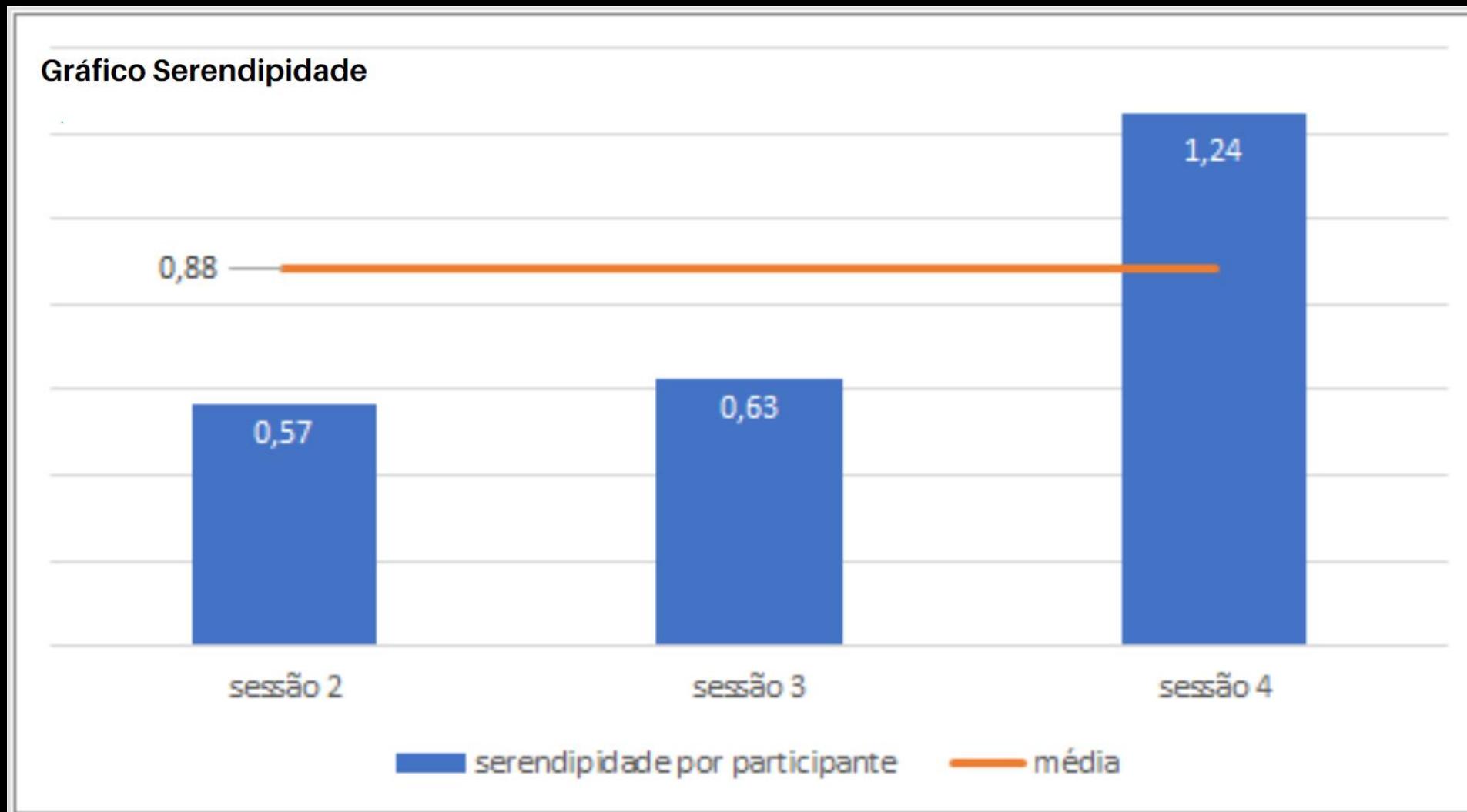
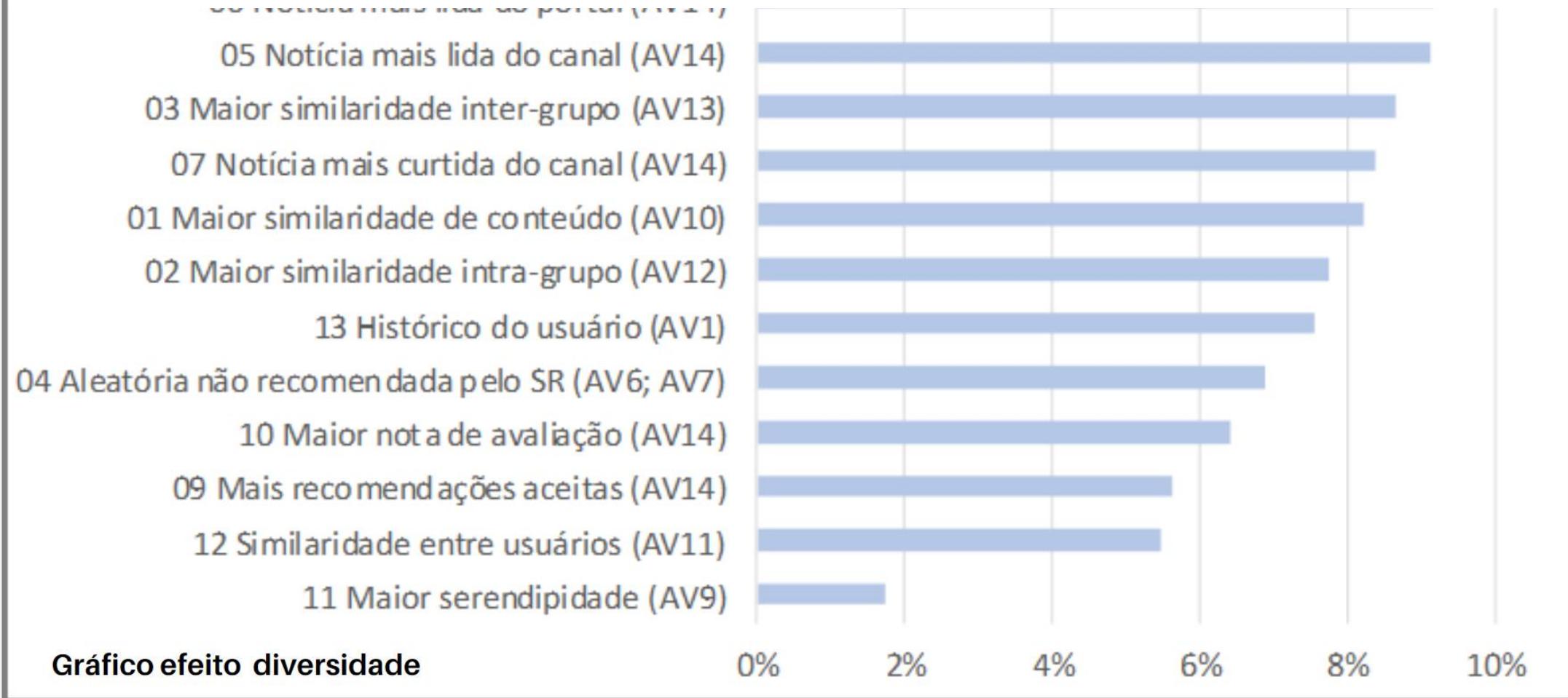


IMAGEM 78 – SERENDIPIDADE POR SESSÃO

Desenvolvendo um Sistema de Recomendação

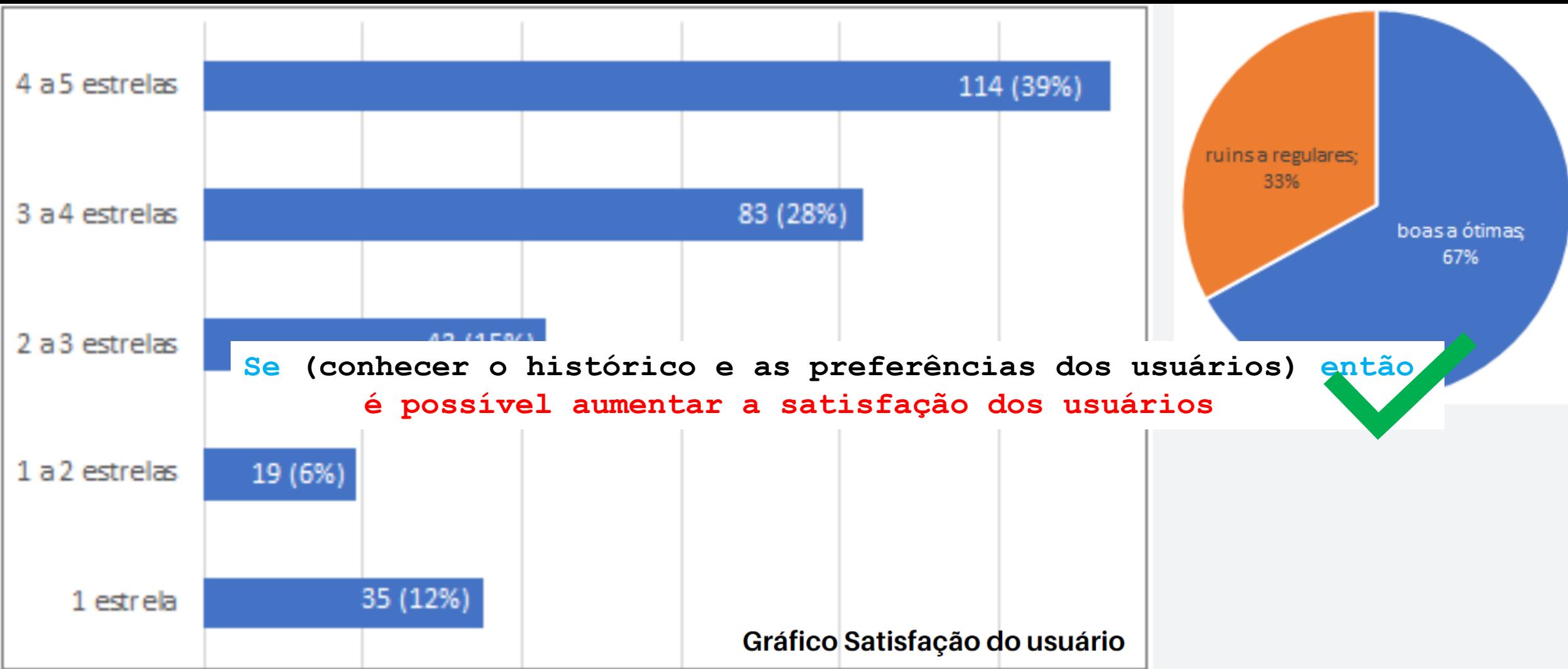
Análise dos dados (Resultados) - verificação da eficácia e confirmação da primeira hipótese

Se (adotar uma abordagem híbrida) então
é possível equilibrar limitações do uso de métodos isolados



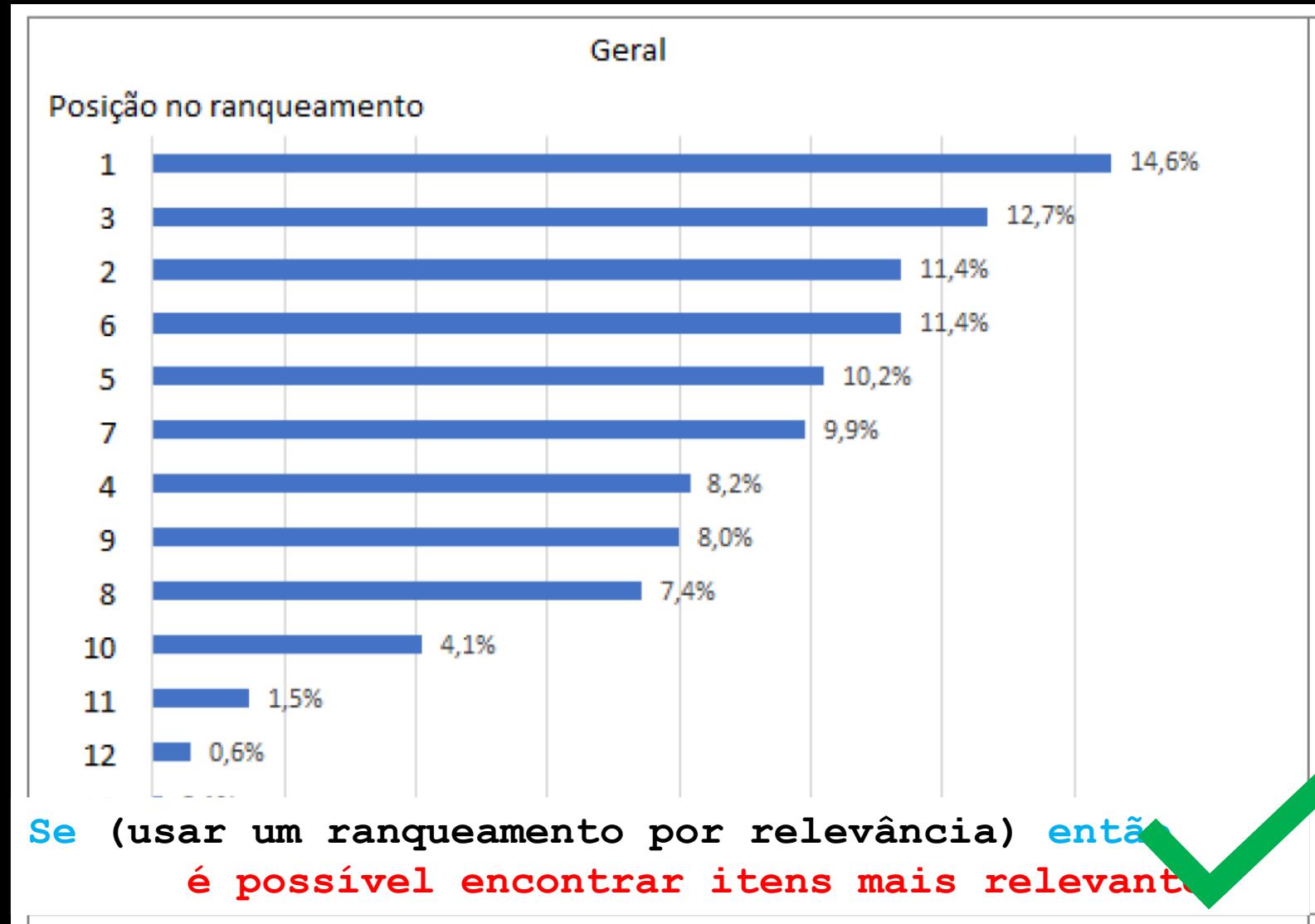
Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados) - verificação da eficácia e confirmação da segunda hipótese



Desenvolvendo um Sistema de Recomendação

Análise dos dados (Resultados) - verificação da eficácia e confirmação da terceira hipótese



Desenvolvendo um Sistema de Recomendação - Dashboard



Capítulo 6

6.2. Tendências dos SRs e considerações finais

SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

- Sistemas de Recomendação podem ser muito “chatos” (**irrelevantes ou inadequados**) sugerindo produtos aleatoriamente ou com base em um histórico de navegação que você já não quer mais consumir.
- Por outro lado são ferramentas muito úteis (**relevantes e adequados**) para usuários finais (cliente/consumidor), para usuários de negócio (que usam o SR como ferramenta de apoio para melhores tomadas de decisões) e ainda pelos proprietários dos SRs (com insights obtidos por análise de dados).
- Neste contexto é importante estarmos atentos às oportunidades de aprimoramento dos SRs, visando proporcionar maior valor agregado ao negócio.

SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

Exemplo recomendações irrelevantes ou inadequadas:



The screenshot shows a news article from the website g1. The title of the article is "Comprou na Black Friday e se arrependeu? Veja o que fazer" (Bought on Black Friday and regretted it? See what to do). The text discusses the long lines and search for televisions during the event. Below the article, there is an advertisement for a Samsung Smart TV.

Anúncio



A Samsung Smart TV advertisement featuring the "Crystal UHD" model. The TV screen displays a vibrant blue water pattern with the text "SAMSUNG 2022 Crystal UHD".

**Smart TV 55" Crystal UHD 4K
Samsung 55BU8000 Painel
Dynamic Crystal Color Design
Slim T...**

[Saiba mais](#)

Americanas.com



The screenshot shows a sidebar on an e-commerce platform. It features four cards with irrelevant recommendations: "bem estar" (well-being), "desenrola, rio" (unwind, river), "prazer, renata" (pleasure, renata), and "bichos na escuta" (insects listening).



The screenshot shows a product listing for a Samsung Smart TV on the Americanas website. The top product is a "Smart TV 55" Crystal UHD 4K" model priced at R\$ 3.449,99. Below it is another product, "Smart TV Samsung 55 LED Cr...", with a price of R\$ 2.496,75. A red banner indicates a 10% discount on the second product.

IMAGEM 83 – RECOMENDAÇÃO IRRELEVANTE E-COMMERCE

SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

Exemplo recomendações irrelevantes ou inadequadas:

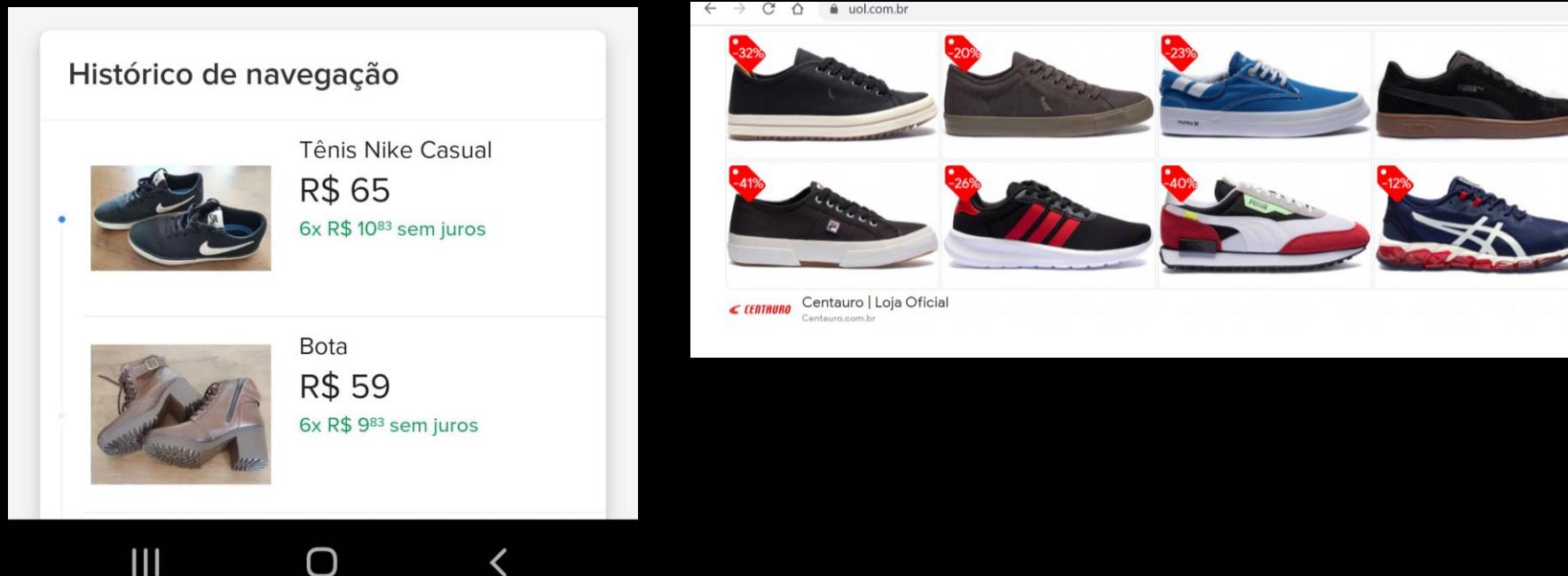
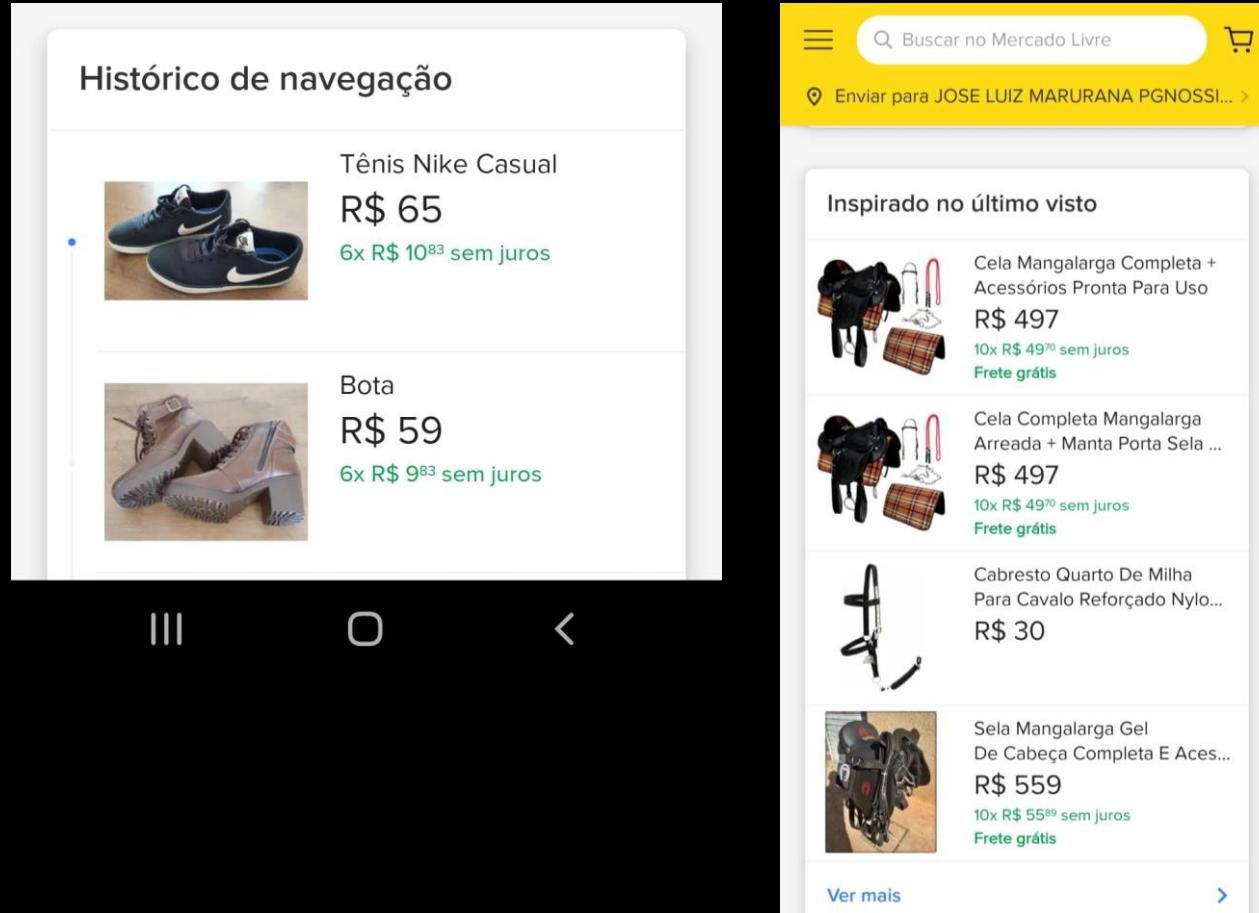


IMAGEM 84 – RECOMENDAÇÃO IRRELEVANTE E-COMMERCE 2

SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

Exemplo recomendações irrelevantes ou inadequadas:



SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

Caso George Floyd: tropas ocupam ruas dos EUA após protestos violentos contra o racismo



pressionou o pescoço da vítima com o joelho durante uma abordagem na última segunda-feira.

RELACIONADAS



De Obama a Kim Kardashian, personalidades protestam por George Floyd



A longa história de segregação e conflito racial em Minneapolis, onde George Floyd foi morto pela polícia



Nike lança manifesto contra o racismo e ganha apoio da Adidas



SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

Caso George Floyd: tropas ocupam ruas dos EUA após protestos violentos contra o racismo



última segunda-feira.

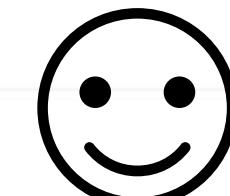
RELACIONADAS



De Obama a Kim Kardashian, personalidades protestam por George Floyd



A longa história de segregação e conflito racial em Minneapolis, onde George Floyd foi morto pela polícia



Nike lança manifesto contra o racismo e ganha apoio da Adidas

Publicidade relacionada com similaridade => maior valor agregado ao negócio



Recomendação fictícia (Oportunidade)

SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

- Estar atento ao lançamento de features geradas por algum tipo de recomendação, como estas:

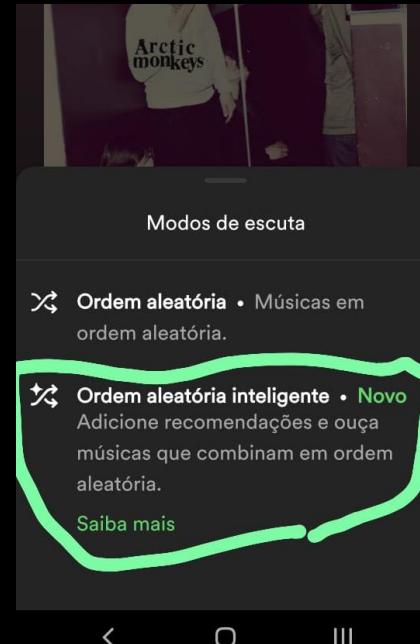


IMAGEM 85 – FEATURES EM SR

SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

- Tornar a recomendação algo natural na experiência do usuário, mesmo não sendo o principal fator de decisão do cliente ao escolher uma plataforma em detrimento de outra, faz com que ele fidelize, se sinta satisfeito e passe cada vez mais tempo na sua plataforma.



IMAGEM 86 – FATORES PREFERENCIA STREAMING

- Fatores explícitos que levam os usuários a preferir uma plataforma:

- Preço
- Acervo
- Performance

- Fatores implícitos que levam os usuários a fidelizar em uma plataforma:

- Usabilidade
- Recomendação

SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

Os SRs tendem a ser cada vez mais:

Híbridos

Proativos

Interativos

Relevantes

Inteligentes

Assistentes e
envolventes



IMAGEM 87 – TENDENCIAS - HER

SISTEMAS DE RECOMENDAÇÃO – TENDÊNCIAS

Apesar de alguns problemas que este tipo de recomendação pode causar, por exemplo:

Privacidade

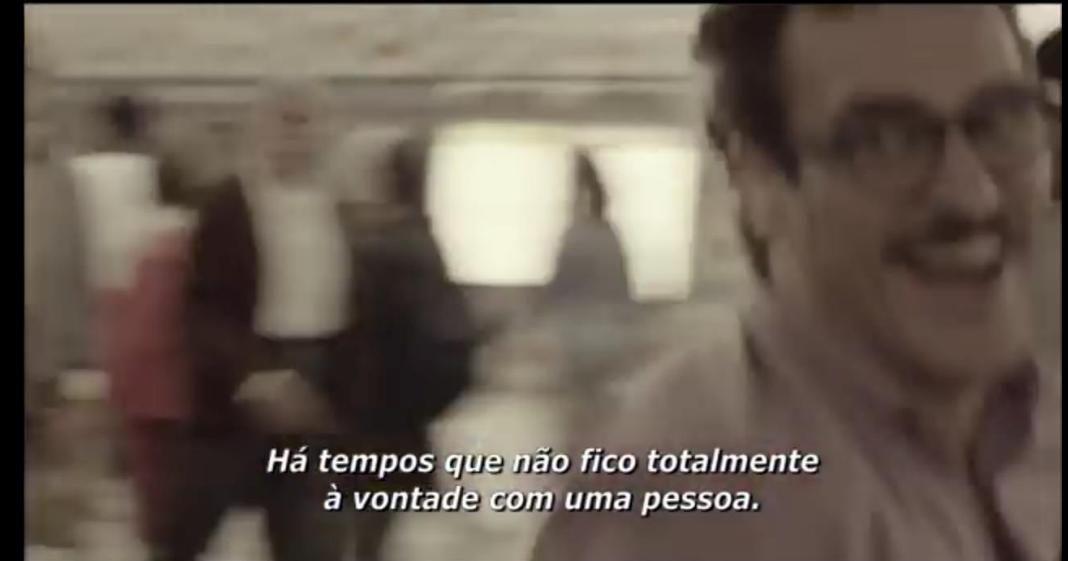
Bolhas sociais

Viés ou manipulação

Isolamento social

Os SRs tendem a ser cada vez mais:

Integrados
naturalmente à
vida das
pessoas



*Há tempos que não fico totalmente
à vontade com uma pessoa.*

IMAGEM 87 – TENDENCIAS SR ENVOLVENTE – HER

SISTEMAS DE RECOMENDAÇÃO – CONSIDERAÇÕES FINAIS

- Iniciar a construção de um SR é relativamente simples, com apenas uma técnica ou controle já se implementa alguma forma de recomendação, como click, timer, contadores ou ainda filtros baseado em algum conhecimento prévio dos dados.
- Por outro lado, vimos também que técnicas isoladas podem trazer problemas ao SR, como Cold-start, aleatoriedade e até previsibilidade;
- Para atacar esses problemas e construir SRs mais robustos e sofisticados, estuda-se propriedades, técnicas e algoritmos. Em posse dessas informações define-se a melhor estratégia para aplicar as ferramentas mais adequadas para cada necessidade, dessa forma utiliza-se então uma abordagem híbrida.

SISTEMAS DE RECOMENDAÇÃO – CONSIDERAÇÕES FINAIS

- Importante também, conhecermos aplicações de mercado e como as principais plataformas da atualidade utilizam esse poderoso recurso que é a recomendação para alavancar seus negócios. Adicionalmente, ter acesso a estudos de casos que implementam SRs é sempre uma boa ideia, pois nos mostra alguns dos possíveis caminhos a seguir, utilizando-se de boas práticas de mercado e acadêmica.
- Por fim, os SRs possuem características particulares em relação ao desenvolvimento de um sistema tradicional, principalmente na fase de Validação, por este motivo, definir as métricas, organizar um experimento, analisar os resultados e confirmar/refutar as hipóteses são essenciais para uma boa avaliação do SR. O resultados podem nos indicar também, necessidade de recalibragem do sistema, lacunas encontradas e oportunidades de trabalhos futuros que devem ser considerados como processo de melhoria contínua do SR.

REFERÊNCIAS

- RICCI, F. et al. Recommender Systems Handbook. [S.I.]: Springer, 2011
- GUNAWARDANA, S. Evaluating recommender systems. In: RICCI, F. e. a. (Ed.). Recommender Systems Handbook Second Edition. [S.I.]: Springer, 2015.
- A. Diaz, A. Lima, A. Silva, F. Costa, J. Pagnossim, S. Peres. Uma análise comparativa das ferramentas de pré-processamento de dados textuais: NLTK, PreTexT e R. Relatórios Técnicos, EACH-USP. São Paulo-SP, 2018.
- PAGNOSSIM, José e Luiz Maturana. Uma abordagem híbrida para sistemas de recomendação de notícias. 2018. 119 f. Dissertação (Mestrado em Ciências) – Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, São Paulo, 2018.
- LAURIA, Rayana de Souza Aplicação de sistemas de recomendação em finanças – FGV, Rio de Janeiro, RJ, 2020.
- Netflix TechBlog. Netflix Recommendations: Beyond the 5 stars (Part 1). 2012. Disponível em: <<https://netflixtechblog.com/netflix-recommendations-beyond-the-5-stars-part-1-55838468f429>>. Acesso em: 26 nov. 2022.

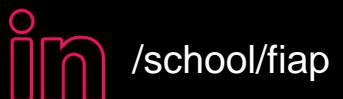
REFERÊNCIAS

- Escola de E-commerce. O motivo por trás do sucesso do Sistema de Recomendação da Amazon. 2020. Disponível em: <<https://www.escoladeecommerce.com/artigos/sucesso-do-sistema-de-recomendacao-da-amazon/>>. Acesso em: 26 nov. 2022.
- EBC. Portal EBC de notícias. 2022. Disponível em: <<https://www.ebc.com.br/>>. Acesso em: 26 nov. 2022.
- iMasters. Principais técnicas de machine learning. 2022. Disponível em: <<https://imasters.com.br/back-end/principais-tecnicas-de-machine-learning/>>. Acesso em: 26 nov. 2022.
- Diogo Nogare. Explicando o algoritmo de regra de associação. 2020. Disponível em: <<https://diegonogare.net/2020/05/explicando-o-algoritmo-de-regra-de-associacao/>>. Acesso em: 26 nov. 2022.
- Santander. Open Finance. 2022. Disponível em: <<https://www.santander.com.br/hotsite/open-finance/>>. Acesso em: 26 nov. 2022.
- Python. Downloads. 2022. Disponível em: <<https://www.python.org/downloads/>>. Acesso em: 26 nov. 2022.
- Hashtag Treinamentos. Como Criar um Gráfico de Dispersão no Excel. 2020. Disponível em: <<https://www.youtube.com/watch?v=IJLKkr4M9ko>>. Acesso em: 26 nov. 2022.
- Warner Bros. Her (Filme). 2020. Disponível em: <<https://www.warnerbros.com/movies/her>>. Acesso em: 26 nov. 2022.
- Youtube. Her - Trailler oficial. 2014. Disponível em: <<https://www.youtube.com/watch?v=hX09Kz7BAIU>>. Acesso em: 26 nov. 2022.

MUITO OBRIGADO
E não deixe de realizar a survey
desta aula no portal FIAP.

 profjose.pagnossim@fiap.com.br

linkedin.com/in/jose-luiz-maturana-pagnossim-3454898



FIAP

Copyright © 2022 | Professor José Pagnossim

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente
proibido sem consentimento formal, por escrito, do professor/autor.