Análise de Subgrupos e Padrões Frequentes na Base de Dados do Stack Overflow 2023

Ester Sara[1], **Filipe Pirola**[1], **Júnio Veras**[1], **Thaís Ferreira**[1], **Vinícius Braga**[1]

[1]Departamento de Ciência da Computação (DCC) -
Universidade Federal de Minas Gerais (UFMG) - Belo Horizonte - MG - Brasil

estersarasilva@gmail.com, filipepirolasantos@hotmail.com,

junioveras127@gmail.com, thaisfdasilva159@gmail.com,

vinciusb15999@gmail.com

Abstract. *This paper employs data mining techniques to uncover patterns and group respondents based on characteristics shared by the Stack Overflow 2023 survey data, emphasizing salary distributions. Several quality measures were used to highlight subgroups with distinct salary tendencys. The methodology includes data collection, preprocessing, frequent pattern mining using the FP-Growth algorithm, and subgroup discovery. The results reveal significant patterns and subgroups, providing valuable insights into salary trends and preferences within the developer community.*

Resumo. *Este artigo emprega técnicas de mineração de dados para descobrir padrões e agrupar os respondentes com base em características compartilhadas pelos dados da base da pesquisa do Stack Overflow 2023, com ênfase nas distribuições salariais. Foram utilizadas várias medidas de qualidade para destacar subgrupos com tendências salariais distintas. A metodologia inclui coleta de dados, pré-processamento, mineração de padrões frequentes utilizando o algoritmo FP-Growth e descoberta de subgrupos. Os resultados revelam padrões e subgrupos significativos, fornecendo resultados valiosos sobre tendências salariais e preferências dentro da comunidade de desenvolvedores.*

1. Introdução

O Stack Overflow, site de perguntas e respostas para programadores, realiza uma pesquisa anual que coleta uma ampla gama de informações sobre desenvolvedores de software, incluindo: Tecnologias favoritas, níveis de experiência, preferências de trabalho e faixas salariais [StackOverflow 2023a]. Com esses dados, esse artigo almeja, utilizando técnicas de mineração de dados, descobrir padrões e agrupá-los com base em características comuns, com um foco especial nas faixas salariais.

O estudo da análise de subgrupos e padrões frequentes é fundamental para entender as dinâmicas dentro da comunidade de desenvolvedores. Ao identificar subgrupos que compartilham características específicas, é possível obter resultados valiosos sobre as tendências salariais e tecnológicas. Por exemplo, ao mapear atributos demográficos e preferências incomuns, os recrutadores podem direcionar suas estratégias para grupos-chave onde determinadas tecnologias ou faixas salariais têm maior potencial.

Table 1. Descrição das colunas da base que foram utilizadas no projeto

Coluna	Descrição
Age	Idade do respondente
Employment	Tipo de ocupação (full-time, estudante, freelancer, ...)
RemoteWork	Remoto, híbrido ou presencial
EdLevel	Maior nível de educação (bacharel, mestre, ...)
YearsCode	Há quantos anos programa
YearsCodePro	Há quantos anos programa profissionalmente
OrgSize	Número de funcionários na empresa
Country	País
DevType	Tipo de desenvolvedor (back-end, mobile, ...)
LanguageHaveWorkedWith	Lista de tecnologias das quais já trabalhou
ConvertedCompYearly	Salário anual convertido em dólares

Na abordagem escolhida, combinamos métodos de descoberta de subgrupos (SD, do inglês, *subgroup discovery*) com medidas de qualidade específicas para avaliar a relevância e a excepcionalidade dos subgrupos descobertos. A metodologia envolve a coleta, pré-processamento e análise da base de dados do Stack Overflow, utilizando algoritmos de mineração de padrões frequentes como o FP-Growth, e técnicas avançadas de SD para revelar subgrupos com padrões salariais distintos (e.g. best-first).

Finalmente, apresentamos uma análise detalhada dos resultados obtidos, destacando os subgrupos e padrões mais significativos. Estes resultados são acompanhados de visualizações que ilustram as tendências descobertas, oferecendo uma visão abrangente sobre as dinâmicas salariais e preferências dentro da comunidade de desenvolvedores. Todo o código fonte do projeto, bem como as imagens contidas nesse artigo podem ser encontrados no GitHub dos autores, identificado nas referências [GitHub 2024].

2. Metodologia

A metodologia empregada neste estudo foi estruturada em várias etapas fundamentais, abrangendo desde a coleta e pré-processamento dos dados até a descoberta de padrões frequentes e de subgrupos. Cada uma dessas etapas foi cuidadosamente planejada para garantir a qualidade e relevância dos resultados obtidos.

2.1. Coleta de Dados

O primeiro passo da metodologia foi a coleta de dados da pesquisa anual de desenvolvedores do Stack Overflow de 2023 [StackOverflow 2023a], que obteve respostas de mais de 90 mil desenvolvedores. Os dados coletados incluíram informações sobre as tecnologias utilizadas, níveis de senioridade e salários anuais como mostra a tabela 1. Esses e mais outros dados fornecem uma base rica para a análise das tendências e padrões dentro da comunidade de desenvolvedores e do mercado de tecnologia.

2.2. Pré-processamento

O pré-processamento dos dados é uma etapa crucial em qualquer análise de dados, garantindo a qualidade e consistência dos dados para a posterior extração de informação útil. A seguir, detalhamos as principais etapas de pré-processamento aplicadas.

2.2.1. Conversão Inicial dos Dados

Algumas conversões de dados foram realizadas para garantir que os tipos de dados estivessem corretos e prontos para análises subsequentes.

A primeira transformação feita foram em relação aos dados de anos de programação que o usuário possuía. Assim, as colunas *YearsCode* e *YearsCodePro* foram convertidos para valores únicos e exclusivamente numéricos (haviam alguns valores que eram textuais, e.g., *50 years or more*).

Outro dado muito crucial que foi necessário tratar é exatamente o *target* da nossa análise: o salário. A base de dados nos fornece o dado do salário anual já convertido em dólares utilizando os câmbios do dia 02/06/2023, como é referenciado na seção de metodologia da base [StackOverflow 2023b]. Sabe-se que a simples conversão de uma moeda para outra não se faz suficiente para nos permitir comparar salários, pois os países envolvidos possuem grandes diferenças econômicas, como por exemplo custo de vida e desvalorização monetária.

Para tornar nossa análise mais precisa, precisamos comparar poderes de compra equiparáveis entre os indivíduos. Para tal, existe a métrica de paridade de poder de compra (PPP, do inglês, *Power Purchase Parity*) [WorldBank 2024]. Utilizamos o PPP em relação aos Estados Unidos da América (EUA), pois havia a conveniência dos salários já estarem convertidos em dólares. Neste caso, a métrica é indexada ao poder de compra dos EUA, cujo valor base é 1. Para se fazer uma conversão de salário (em dólar) para poder de compra compatível com um salário americano, basta aplicar a formula abaixo:

$$Salario_{Pareado} = PPP * Salario_{ConvertidoEmDolar} \tag{1}$$

2.2.2. Tratamento de *Outliers*

Os *outliers* foram tratados removendo registros cujo salário desviavam para mais ou menos de 3 desvios padrões em relação a média da base. Além disso, filtrou-se os dados para incluir apenas países com um número significativo de respostas, resultando em um grupo de 27 países relevantes (os 27 mais populosos na *survey*). A remoção de *outliers* garantiu que valores extremos não distorcessem a análise, enquanto a inclusão apenas de países com muitas respostas assegurou a representatividade dos dados.

2.2.3. *One-Hot Encoding*

Dada as listas de tecnologias (e.g., Python, Lua, ...) da coluna *LanguageHaveWorkedWith* foi aplicado o *one-hot encoding* para converter essas varias variáveis categóricas em numéricas através de uma representação binária onde cada coluna representa uma tecnologia e cada linha tem um valor de 0 ou 1 indicando a ausência ou presença da tecnologia na lista de tecnologias de conhecimento do participante. No total, haviam 51 tipos diferentes de linguagens e, portanto, 51 novas colunas foram adicionadas.

2.3. Análise de Correlação

Inicialmente, consideramos agrupar colunas com alta correlação para analisar os impactos das variações entre opções semelhantes. No entanto, devido à baixa correlação entre a maioria das colunas, essa abordagem foi descartada. As exceções encontradas nesse processo já eram esperadas, não acrescentando nada de relevante ao projeto, como a relação entre HTML e JavaScript para desenvolvimento web.

Outro fator importante a se salientar é que a correlação entre as colunas descritoras e a coluna alvo também são muito baixas, deixando claro que os padrões a serem encontrados não poderiam ser facilmente encontrados por uma simples indução estatística.

2.4. Análise de Subgrupos

Análise de Subgrupos é uma técnica de mineração de dados que visa identificar e caracterizar grupos específicos dentro de um conjunto de dados onde uma variável alvo exibe uma distribuição incomum em comparação com a população geral. Para sua aplicação, devemos pré processar os dados de forma a garantir consistência e qualidade, selecionar um algoritmo de busca (e.g. CN2-SD, Apriori-SD, SD-Map) e definir medidas de qualidade a serem usadas para avaliar os subgrupos escolhidos (e.g. WRAcc).

Para realizar os experimentos com diferentes estratégias, escolhemos o Cortana, ferramenta utilizada para encontrar e analisar subgrupos dentro de um conjunto de dados que disponibiliza diversas implementações e configurações para algoritmos de descoberta de subgrupo. A seção irá apresentar os vários aspectos escolhidos, como estratégias de busca e formulas de qualidade.

Primeiro foram feitos alguns tratamentos de dados menores, como remover colunas, dados nulos e espaços de campos textuais (uma vez que a ferramenta não consegue lidar bem com colunas que tenham *white spaces*).

Feito isso, extraiu-se os quartis da base para se analisar de forma mais apurada as extremidades da base (i.e. 1° e 4° quartil). Analisamos somente tais quartis pois eles são capazes de nos gerar subgrupos cuja distribuição variando para cima ou para baixo nos dá mais informação sociais (pois estamos lidando das parcelas mais ricas/pobres desta população), como por exemplo casos de salários extremamente baixos ou extremamente altos.

Definindo as configurações do Cortana para aplicar a mineração de subgrupos, teremos:

- ***Target Type:*** Refere-se ao tipo de variável ou atributo que se deseja analisar ou prever dentro de um conjunto de dados. No conjunto em questão, o foco é em variáveis numéricas, afinal nossa variável alvo é o poder de compra.
- ***Quality Measure:*** Critério utilizado para avaliar a relevância, qualidade e utilidade dos subgrupos identificados. No nosso caso, utilizamos:
  - **Z-score:** O Z-score é calculado com base na média e no desvio padrão do conjunto de dados. O cálculo do Z-score para um determinado subgrupo envolve comparar a frequência observada desse subgrupo (O) com a frequência esperada (E), que é baseada no desvio padrão (σ).

$$Z = \frac{O - E}{\sigma} \tag{2}$$

– **Weighted Kullback-Leibler (KL):** É uma extensão da divergência KL (mede o quanto uma distribuição de probabilidade (P) diverge de uma segunda distribuição de probabilidade (Q) que incorpora pesos para ajustar a importância das diferenças observadas entre subgrupos e a distribuição geral. Os pesos ($w(x)$) podem refletir a importância ou a frequência relativa dos diferentes elementos ou subgrupos no conjunto de dados.

$$D_{KL}^{W}(P||Q) = \sum_{x} w(x) * P(x) log \frac{P(x)}{Q(x)} \tag{3}$$

- **Search Strategy Type:** Estratégia de busca escolhida, estamos usando a Best-First (Algoritmo 1).

Algorithm 1 Busca Best-First

Require: Nó inicial s, nó objetivo g, função heurística h
Ensure: Caminho de s para g (se encontrado)
  Inicializar lista aberta com s
  Inicializar lista fechada como vazia
  while a lista aberta não estiver vazia **do**
    $n \leftarrow$ nó na lista aberta com o menor valor de h
    if n é o nó objetivo **then**
      return caminho do nó inicial até n
    end if
    Remover n da lista aberta
    Adicionar n à lista fechada
    for cada vizinho m de n **do**
      if m está na lista fechada **then**
        continuar
      end if
      if m não está na lista aberta **then**
        Adicionar m à lista aberta
      end if
      Definir o pai de m como n
    end for
  end while
  return falha (se nenhum caminho for encontrado)

2.5. Mineração de Padrões

Mineração de padrões frequentes é uma técnica usada para encontrar padrões comuns em grandes conjuntos de dados, ou seja, nessa aplicação isso envolve identificar combinações de tecnologias e características dos desenvolvedores que ocorrem frequentemente juntos utilizando o FP-Growth. Neste exemplo, cada item será uma descrição do indivíduo (e.g. já trabalhou com Python, é desenvolvedor full-stack, ...) e o *itemset* será o conjunto de características que o compõe.

Para tal atividade, realizamos uma filtragem a mais no poder de compra, mantendo apenas as linhas onde esse valor é menor que 2 milhões (uma remoção de *outliers* além da descrita anteriormente, pois a calda superior de salários estava muito alongada).

	Coverage	Quality	Average	St. Dev.	Conditions
0	489	0.022808	5.362841	0.115212	Country = 'India'
1	432	0.020876	5.366105	0.114653	Country = 'India' AND Employment = 'Employed-full-time'
2	97	0.019173	5.474681	0.064695	Country = 'Sweden'
3	379	0.018062	5.360752	0.117345	Country = 'India' AND EdLevel = 'Bachelor's-degree-(B.A.-B.S.-B.Eng.-etc.)'
4	83	0.017696	5.482628	0.051838	Country = 'Sweden' AND Employment = 'Employed-full-time'

Figure 1. Subgrupos encontrados na análise sem colunas de anos de programação

Depois realizamos a criação de 4 quartis de acordo com o poder de compra dos desenvolvedores. E também variamos as colunas para utilização do algoritmo de mineração de padrões frequentes em 3 casos: dados demográficos, tecnologias e linguagens de programação e uma junção dos dois . Após isso, criamos um mapeamento de um valor único para cada item em nossa base (e.g.: python mapeia para 5, C# para 120) para que ficasse no formato necessário para utilização no algoritmo de padrões frequentes.

Por fim, utilizamos o algoritmo FP-Growth implementado na biblioteca SPMF (Sequential Pattern Mining Framework) [Fournier-Viger 2024] para identificar os padrões frequentes com um suporte mínimo de 30%.

3. Resultados

Após realizar a mineração dos dados, analisamos os resultados obtidos. Na primeira seção, conferimos os subgrupos descobertos e, em seguida, discutimos os padrões frequentes encontrados.

3.1. Subgrupos

Inicialmente, todos os resultados apresentados não consideraram as colunas de stacks (colunas resultantes do *one-hot encoding*). Essas colunas foram desconsideradas, pois durante todos os experimentos das quais foram incluídas não deram bons resultados. Além disso, as condições de busca, *refinement depth* e *maximum subgroups*, foram definidas com valores 7 e 5, respectivamente, durante todos os experimentos.

3.1.1. 4° Quartil - Suécia e Índia

Neste exemplo estamos trabalhando no 4º quartil. Para esta análise, utilizamos duas bases de dados distintas. A primeira incluiu as colunas *YearsCode* e *YearsCodePro*, enquanto a segunda desativou essas colunas. Vamos apresentar os resultados do segundo caso, pois o primeiro apenas reforçou o padrão "Suécia" que apresentaremos a seguir, enquanto o segundo não só apresentou esse comportamento, mas também outros subgrupos com maior diversidade (Fig. 1).

Podemos perceber que, para os subgrupos da Índia, os salários possuem média maior que a média populacional do 4º quartil (pessoas que recebem os 25% salários mais altos, cuja média é $5,29$). Isto ocorre principalmente para profissionais com ensino superior ou empregados em tempo integral.

Esse comportamento passa a fazer sentido quando observamos que os Indianos estão entre os melhores programadores do mundo. Se observamos bem, as maiores *big techs* do mundo possuem CEOs indianos, mostrando a qualidade que esses desenvolvedores possuem, principalmente em cargos *high end* (cargos de maior senioridade e importância), que são exatamente os cargos representados pelo 4º quartil.

Em adição, foi notado nas duas bases deste caso de uso a Suécia foi o país que mais se destaca no 4° quartil quando estamos falando em receber acima do restante da população. Ao analisarmos a figura 2 e a linha 3 da figura 1 vemos que existe um subgrupo de suecos que recebem bem além do que o restante do 4° quartil.

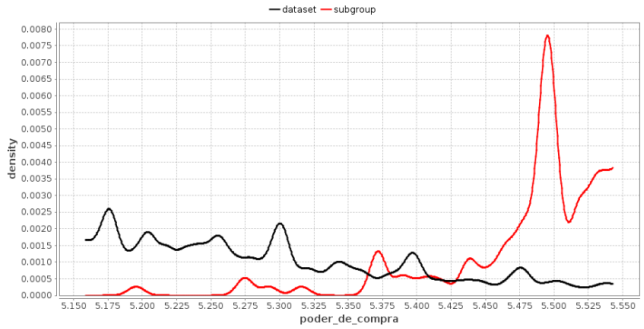

Figure 2. Curva de distribuição da população do subgrupo [Country = ”Sweden”]

Uma suposição que explica este comportamento é que a Suécia tem melhores condições de trabalho que outros países como os Estados Unidos, o que impede que haja um achatamento dos salários no país devido ao uso de mão de obra estrangeira/remota mais barata.

3.1.2. 4° Quartil - Mestrado

Neste teste, ao contrário do anterior que filtrou os dados para salários abaixo de $\$350000$, neste utilizamos salários abaixo de $\$2000000$. As colunas *YearsCode* e *YearsCodePro* manteram-se desativadas. No entanto, para buscar padrões diferentes dos convencionais, também desativamos a coluna *Country*. Como essa coluna tem alta dominância sobre as demais, realizar alguns testes sem considerá-la pode ajudar a entender quais padrões o algoritmo consegue captar de forma geral (Fig. 3). Por fim, continuamos utilizando a métrica Z-Score.

Podemos observar pelas distribuições de população das linhas 1, 2, 4, 5 da figura 3 que pessoas com mestrado tem um salário maior que a média $(5,54)$ dos maiores salários

	Coverage	Quality	Average	St. Dev.	Conditions
0	1137	12.620914	5.639455	0.254283	RemoteWork = 'Hybrid-(some-remote-some-in-person)' AND EdLevel = 'Master's-degree-(M.A.-M.S.-M.Eng.-MBA-etc.)'
1	1137	12.620914	5.639455	0.254283	EdLevel = 'Master's-degree-(M.A.-M.S.-M.Eng.-MBA-etc.)' AND RemoteWork = 'Hybrid-(some-remote-some-in-person)'
2	3899	12.447682	5.596046	0.241376	RemoteWork = 'Hybrid-(some-remote-some-in-person)'
3	1167	12.099188	5.634464	0.251312	Age = '25-34-years-old' AND EdLevel = 'Master's-degree-(M.A.-M.S.-M.Eng.-MBA-etc.)'
4	1167	12.099188	5.634464	0.251312	EdLevel = 'Master's-degree-(M.A.-M.S.-M.Eng.-MBA-etc.)' AND Age = '25-34-years-old'

Figure 3. Subgrupos encontrados na análise 3

da área de TI (4° quartil). As figuras 4 e 5 mostram, respectivamente, as curvas de distribuição dos subgrupos das linhas 2 e 5 da figura 3.

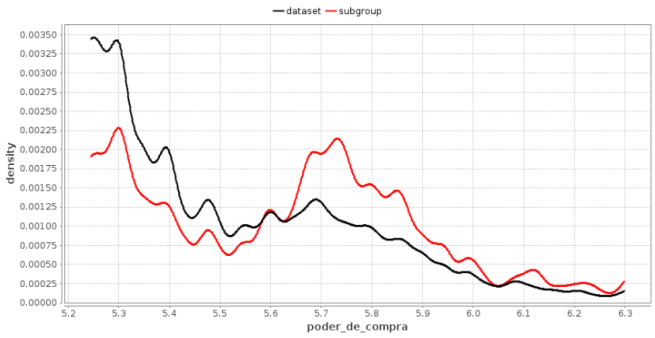

Figure 4. Curva de distribuição da população da linha 2

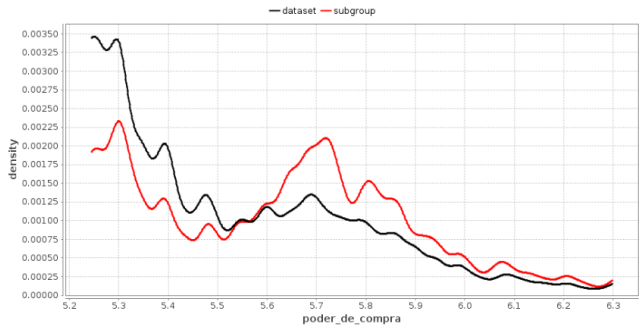

Figure 5. Curva de distribuição da população da linha 5

Este padrão ajuda a quebrar um pouco do estigma que existe no Brasil e, talvez, em outros países, de que o mestrado não se paga ao longo prazo quando se comparado com se seguir uma carreira profissional no mercado privado. Este dado pode nos dar uma perspectiva diferente sobre isto.

3.1.3. 4° Quartil - Trabalho remoto abaixo entre os mais ricos

Neste teste passamos a utilizar a métrica de qualidade *Weighted Kullback-Leibler* para uma população com salário abaixo de $2000000. Neste caso somente as colunas *YearsCode* e *YearsCodePro* foram desativadas. Vale ressaltar também que não utilizamos escala logarítmica, porém z-normalizamos os dados neste caso, logo a média é 0.

Como podemos ver na figura 6, todas linhas estão abaixo da média populacional e todos eles incluem o fator do trabalho remoto. Por exemplo, podemos ver na figura 7 a

	Coverage	Quality	Average	St. Dev.	Conditions
0	4480	0.022522	-0.157542	1.012564	RemoteWork = 'Remote'
1	3469	0.018439	-0.187554	0.973169	RemoteWork = 'Remote' AND Employment = 'Employed-full-time'
2	2417	0.017198	-0.233504	0.948633	RemoteWork = 'Remote' AND EdLevel = 'Bachelor's-degree-(B.A.-B.S.-B.Eng.-etc.)'
3	1968	0.015722	-0.262065	0.907054	RemoteWork = 'Remote' AND Employment = 'Employed-full-time' AND EdLevel = 'Bachelor's-degree-(B.A.-B.S.-B.Eng.-etc.)'
4	1968	0.015722	-0.262065	0.907054	RemoteWork = 'Remote' AND EdLevel = 'Bachelor's-degree-(B.A.-B.S.-B.Eng.-etc.)' AND Employment = 'Employed-full-time'

Figure 6. Subgrupos encontrados na análise 3

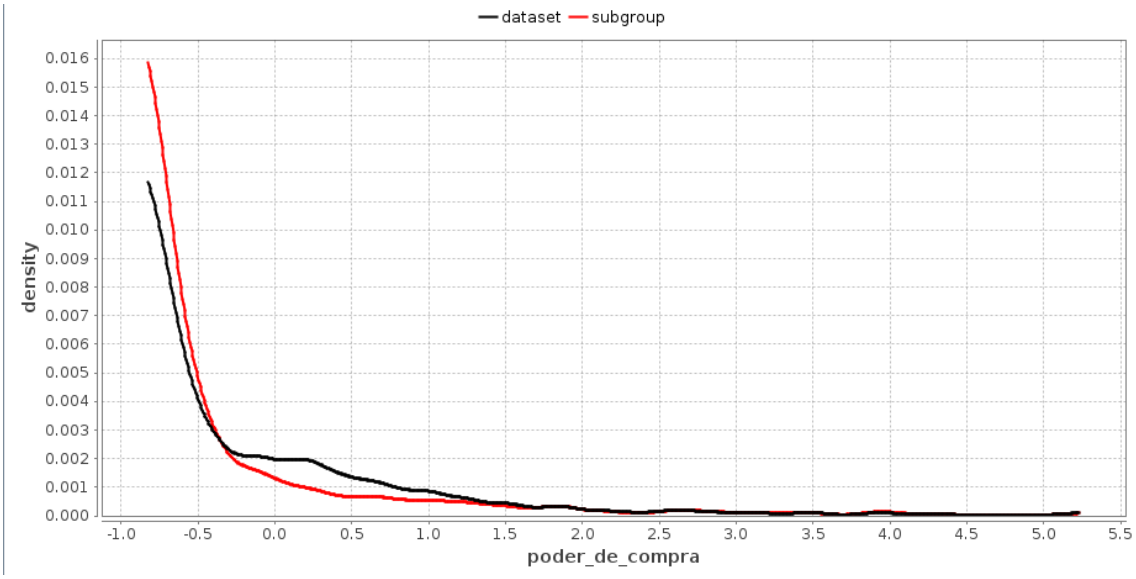

Figure 7. Curva de densidade de salário de subgrupo remoto no 1° quartil

	Coverage	Quality	Average	St. Dev.	Conditions
0	1441	0.038532	0.573701	0.804433	Country = 'Germany' AND Employment = 'Employed-full-time'
1	1735	0.037025	0.510395	0.865310	Country = 'Germany'
2	889	0.027594	-0.488827	0.794451	Country = 'Italy'
3	756	0.027056	-0.509148	0.754766	Country = 'Italy' AND Employment = 'Employed-full-time'
4	865	0.026219	0.612604	0.741774	Country = 'Germany' AND Age = '25-34-years-old' AND Employment = 'Employed-full-time'

Figure 8. Subgrupos encontrados em análise do 1° quartil

distribuição da linha 1, mostrando que pessoas que trabalham remoto recebem abaixo da média (dos mais ricos, i.e., 4° quartil) caso estejam dentre os maiores salários.

Este padrão passa a fazer sentido quando lembramos que há uma tendência global entre os países mais desenvolvidos de se contratar profissionais de alta qualidade de países sub-desenvolvidos por um salário muito menor. A maioria desses casos são pessoas de alta senioridade, se enquadrando entre os maiores salários, porém recebendo menos que pessoas que trabalham presencialmente, em um mesmo cargo, nesses países mais ricos.

3.1.4. 1° Quartil - Itália e Alemanha

Até este ponto achamos subgrupos apenas na porção mais rica dos desenvolvedores. Vamos observar um pouco mais o 1° quartil (pessoas com os menos salários). Neste caso iremos trabalhar com os dados z-normalizados e com as colunas de *YearsCode* e *YearsCodePro* desativadas. Novamente a métrica de qualidade *Weighted Kullback-Leibler* foi utilizada.

Como podemos observar na figura 8 que temos subgrupos com média acima e abaixo da média populacional ($0).

Na figura 9 vemos a distribuição do subgrupo da linha 1. Vemos que dentre os desenvolvedores que menos recebem, ser alemão te da uma maior chance de ter uma

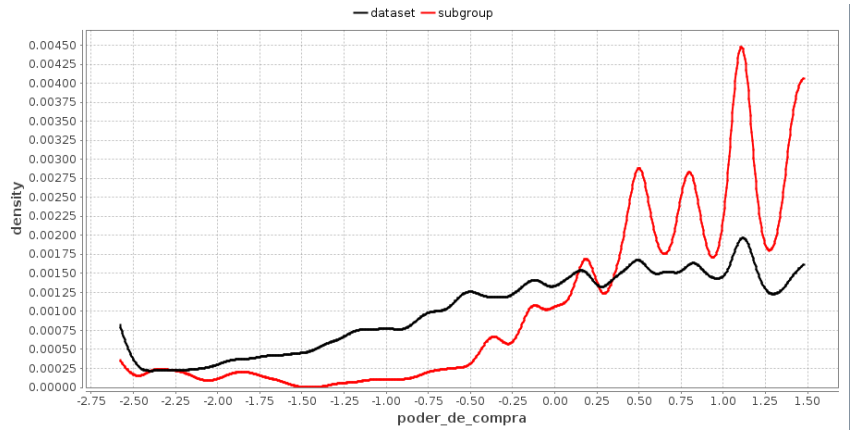

Figure 9. Distribuição do subgrupo da Alemanha

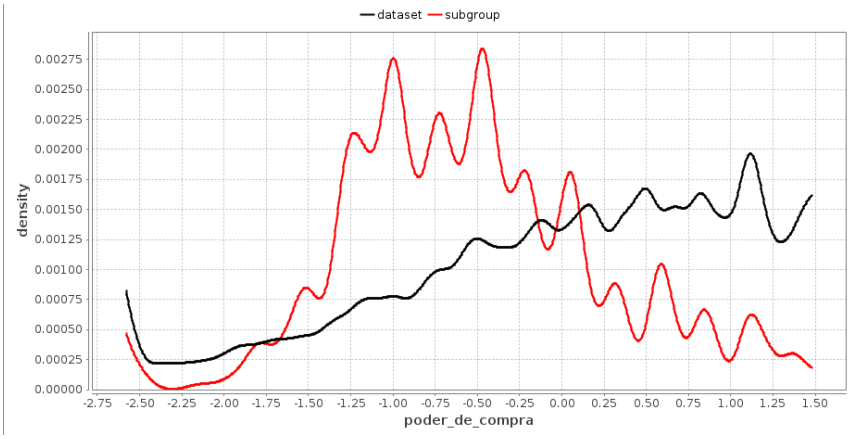

Figure 10. Distribuição do subgrupo da Itaia

média salarial maior. Isso pode ser interpretado por causa das rigorosas leis trabalhistas no país, que garantem que a massa de trabalhadores que estejam próximos do salário mínimo tenham condições melhores que em outros países.

Já na figura 10 vemos a distribuição do subgrupo da linha 4. Vemos que dentre os mais pobres dos desenvolvedores, ser italiano te dá uma maior chance a receber menos que a média. Isso pode advir do fato que a Itália não tem um parque de software tão forte como outros países nesta lista (EUA, India, Alemanha). A situação financeira do país também pode explicar essa situação.

3.2. Padrões Frequentes

Nesta seção, apresentamos os padrões frequentes mais relevantes encontrados nos dados. Das 3 análises realizadas somente 1 encontrou resultados significativos e não redundantes. Os demais resultados podem ser consultados no notebook disponibilizado [GitHub 2024].

3.2.1. Tecnologias e linguagens de programação

1. **Redis**: Este banco de dados aparece exclusivamente nos dois últimos quartis das faixas salariais, sugerindo que a demanda por otimização de desempenho e

serviços com cache está predominantemente associada a posições de maior senioridade. Isso pode refletir a necessidade de soluções avançadas e eficientes em cargos mais experientes.

2. **Google Cloud**: O uso do Google Cloud é restrito ao quartil superior, indicando que profissionais que utilizam esta plataforma são geralmente mais bem remunerados. Essa tendência pode apontar para a valorização de habilidades especializadas em plataformas de nuvem de ponta, frequentemente associadas a cargos de alta responsabilidade, como engenheiros de dados e especialistas em DevOps.
3. **Javascript/HTML/CSS**: Essas tecnologias estão presentes em todas as faixas salariais, evidenciando a onipresença do desenvolvimento web. A popularidade dessas tecnologias reflete a alta demanda por profissionais de desenvolvimento web, uma vez que uma grande parte das oportunidades de emprego está relacionada a essas competências. Vale destacar que o JavaScript é amplamente utilizado tanto em stacks de front-end, em combinação com HTML e CSS, quanto em stacks de back-end, frequentemente associado a bancos de dados como MySQL.
4. **Go**: A linguagem de programação Go é observada apenas no quartil superior. Essa observação está alinhada com o crescente interesse por Go devido à sua eficácia em sistemas de alta performance e escalabilidade, o que tende a atrair cargos mais especializados e, consequentemente, mais bem remunerados.
5. **Azure e AWS**: Essas plataformas de nuvem são comuns em todos os níveis salariais, aparecendo em diversas combinações. Em contraste, o Google Cloud foi encontrado de forma isolada, o que pode sugerir uma adoção mais variada e difundida de Azure e AWS em diferentes faixas salariais, refletindo sua popularidade e aplicação generalizada no mercado.

4. Conclusão

O primeiro objetivo desse artigo foi, com base em técnicas de mineração de padrões e descoberta de subgrupos, explorar a base de dados de pesquisa do Stack Overflow de 2023 para agrupá-los com base em características comuns, com um foco especial nas faixas salariais, buscando encontrar resultados significativos a serem usados para entender tendências salariais e de mercado.

Seguindo na descrição das análises realizadas, esse artigo focou em dois tipos diferentes de abordagem: Descoberta de Subgrupos e Mineração de Padrões. Na primeira, buscou-se encontrar grupos específicos onde uma variável alvo exibe uma distribuição incomum com a utilização da ferramenta Cortana. Foram testadas duas diferentes métricas de qualidade: Z-score e Weighted Kullback-Leibler (KL), além de ter sido definido o uso do Best-First como algoritmo de busca. Por fim, na abordagem de Mineração de Padrões Frequentes, buscamos identificar combinações de linguagens dos desenvolvedores que ocorrem frequentemente juntas utilizando o algoritmo de FP-Growth.

Na seção de resultados, com base nas análises feitas na etapa de metodologia, discutimos sobre as descobertas realizadas. No caso da descoberta de subgrupos, fomos capazes não só de descobrir subgrupos inusitados (vide Suécia ou mestrado), mas descobrir também tendências do mercado de desenvolvimento como um todo.

No caso da Mineração de Padrões, a análise de linguagens de programação em relação a faixa salarial mostrou-se a mais relevante, encontrando padrões interessantes,

como as ferramentas Javascript/HTML/CSS e sua onipresença em todas as faixas salariais, enquanto o Google Cloud, Go e outros estão atrelados a cargos de maior remuneração devido a complexidade/demanda dos mesmos.

Por fim, um trabalho futuro no campo poderia focar em descobrir se os subgrupos encontramos se restringem somente a área de desenvolvimento. Outro ponto interessante seria expandir essa exploração para uma base mais abrangente e rica de desenvolvedores, que poderiam ser fornecidas por sites como Glassdoor e LinkedIn.

References

Fournier-Viger, P. (Jun 12, 2024). Spmf an open-source data mining library v2.62. https://www.philippe-fournier-viger.com/spmf/index.php.

GitHub (Jul, 2024). An alise de subgrupos e padr oes frequentes na base de dados do stack overflow 2023. https://github.com/vinciusb/TP-AD.

Singh, S. (Dez 21, 2023). Best first search (informed search). https://www.geeksforgeeks.org/best-first-search-informed-search/.

StackOverflow (2023a). Stack overflow annual developer survey. https://survey.stackoverflow.co/2023/.

StackOverflow (2023b). Stack overflow annual developer survey methodology. https://survey.stackoverflow.co/2023/#methodology.

WorldBank (2024). Ppp conversion factor, gdp (lcu per international $). https://data.worldbank.org/indicator/PA.NUS.PPP?end=2023&skipRedirection=true&start=1990&view=chart.