Universidade Federal do Rio de Janeiro
Departamento de Ciência da Computação

Data Warehousing no Suporte à Tomada de Decisão - Relatório Individual

Aluna: Domenica Cioci (115089044)
Professor: Geraldo Xexéo

Novembro
2021

Universidade Federal do Rio de Janeiro
Departamento de Ciência da Computação

Prova Individual

Relatório da prova individual, requisito parcial para a aprovação na disciplina de Data Warehousing no Suporte à Tomada de Decisão do Bacharelado em Ciência da Computação.

Aluna: Domenica Cioci (115089044)

Professor: Geraldo Xexéo

Novembro
2021

Conteúdo

1 Extração

Os dados do trabalho foram obtidos diretamente pelo site do INEP, no link Microdados do Exame Nacional de Desempenho dos Estudantes.

Foi utilizado um script em *Python*, localizado no *GitHub* no link 01_download.py.

Os arquivos baixados seguem divididos nas pastas referentes ao seu respectivo ano, seguindo a estrutura:

1. LEIA-ME
    - Dicionário de variáveis dos microdados

      As tabelas contendo a descrição e tipos de cada coluna dos microdados, no formato *.ods* e no formato *.xlsx*
    - Manual do usuário

      Arquivo em *.pdf* contendo a descrição dos arquivos contidos no *.zip* "Microdados"e a motivação para a criação de tais arquivos
    - Questionário do Estudante

      Arquivo em *.pdf* que contém as perguntas aplicadas aos estudantes naquele respectivo ano, cujo índice e resposta consta no *csv* dos microdados
2. INPUTS
    - Programa na linguagem R que lê a tabela em csv
    - Programa em SAS que lê a tabela csv dos microdados, listando o label de suas linhas
    - Programa em SPS que lê a tabela csv dos microdados, colocando o label de suas linhas
3. DADOS
    - Arquivo csv contendo a tabela dos microdados do ENADE do ano em questão

      A tabela possui os campos de identificação do estudante, da IES e do curso, a descrição e o gabarito das questões e as notas obtidas, além da descrição e resposta das questões objetivas contidas no Questionário do Estudante

2 Modelagem Dimensional

Na modelagem, a tabela fato escolhida foi o próprio exame, já que cada exame possui uma dimensão que pode ser obtida da combinação das colunas dos microdados. Outra opção seria utilizar a instância da prova como tabela fato, mas não encontrei uma alternativa que não resultasse no uso do modelo floco de neve.

A primeira versão do modelo não considerava a tabela PROVA, mas foi necessária a decomposição da tabela Q_OBJ nas tabelas Q_OBJ e P_PROVA, já que os campos CO_RS_I1, CO_RS_I2, CO_RS_I3, CO_RS_I4, CO_RS_I5, CO_RS_I6, CO_RS_I7, CO_RS_I8 e CO_RS_I9 não eram parte da prova objetiva, apesar de serem questões objetivas, e sim questões sobre a percepção do estudante sobre a prova. Uma análise da quantidade de questões objetivas do exame, por exemplo, não deveria ser atrelada às questões sobre a prova em si, então algumas mudanças se mostraram necessárias.

Outra alteração do primeiro modelo em relação ao modelo final foi que as colunas CO_TURNO_GRADUACAO e CO_MODALIDADE foram movidas para a tabela de ALUNOS: um curso possui apenas um município, uma região e uma unidade federativa a qual ele pertence - ou seja, apenas uma localidade, mas, com o campo CO_TURNO_GRADUACAO e CO_MODALIDADE, temos que um curso pode ter vários turnos e mais de uma modalidade. Por outro lado, um aluno só pode estar atrelado a um turno e uma modalidade (presencial ou EAD), então a alteração evita registros duplicados na tabela CURSO.

Optei por criar a tabela PRESENCA, que registra os diferentes tipos de presença ligados a cada exame, que antes ficavam como atributos da própria tabela EXAME. Um exame só possui um tipo de presença, então não ocorreria a duplicação de dados nesse caso, porém ao criar uma nova tabela, obtenho uma granularidade maior do banco.

O segundo modelo foi utilizado para modelar o banco, após as mudanças necessárias.

O software utilizado para o desenvolvimento do modelo foi o draw.io, software online integrado com o Google Drive, e optei por exportar o diagrama em .png. As duas versões do modelo estrela se encontram na seção 6.

3 O banco de dados relacional

3.1 Criação das tabelas

A criação das tabelas foi feita com a função *create_tables* no script 02_create.py, também no *GitHub*.

Foi utilizado o módulo *sqlite*3 da biblioteca padrão para executar *querys* diretamente no banco.

O tipo de cada coluna foi obtido analisando o arquivo .sas, utilizando o Visual Studio Code e o dicionário das variáveis, aberto com o Excel Online.

Foram criadas 12 tabelas, além da tabela fato, de acordo com a modelagem dimensional.

3.2 Inserção nas tabelas

A inserção das colunas nas tabelas foi feita com a função *insert_rows_into_db* no script 03_insert.py, utilizando a classe *csv.DictReader* do módulo *csv* da biblioteca padrão para ler e parsear os arquivos .csv dos microdados. Dessa forma, a carga de dados foi feita realizando as *querys* direto para o banco.

A tabela do ano de 2017 possui colunas a mais referentes ao questionário aplicado aos alunos de cursos de licenciatura. Essas colunas não constam nos anos de 2018 e 2019, tornando necessário normalizar as tabelas. Foi feita a substituição de espaços sem valor, espaços contendo ' ' e espaços de tipo numérico contendo 'NA' por *null*.

Para gerar as chaves primárias sequenciais das tabelas, para cada linha lida foi retornado o id sequencial com o atributo *lastrowid* do módulo *sqlite*.

3.3 Adaptando às convenções de nome

Apesar de ter havido um esforço em manter os nomes de acordo com a modelagem em 2, os nomes das colunas ficariam demasiadamente grandes, apesar de mais descritivos. Então algumas tiveram seus nomes encurtados, com o esforço de mantê-los de forma que, ao ler, fosse possível saber do que se tratavam.

4 Análise de dados

Após a criação do banco, optei por continuar utilizando o módulo *sqlite*3 para realizar as consultas no banco e utilizar a biblioteca *Matplotlib* para plotar os gráficos correspondentes.

Após carregada a base de dados, escolhi plotar os gráficos referentes às seguintes perguntas:

- Qual é a distribuição das notas no ENADE em relação ao gênero dos candidatos?
- Qual a distribuição das notas em relação à raça dos candidatos?
- Qual a média das notas em relação às regiões?
- Qual a porcentagem de cada grupo de renda média por região dos inscritos?
- Qual a porcentagem de cada política de ação afirmativa por região?

As análises e gráficos foram gerados com o script 04_analysis.py. Esse script precisa das bibliotecas *Matplotlib* e *Numpy* instaladas para poder ser executado e gera, para cada pergunta, os seguintes arquivos:

- Primeira pergunta:
  - genero-nota.csv: tabela que contém a relação de gênero e nota para todos os candidatos
  - genero-nota.png: gráfico box-plot com a distribuição das notas para cada gênero
- Segunda pergunta:
  - raca-notas.csv: tabela que contém a relação de raça e nota para todos os candidatos
  - raca-notas.png: gráfico box-plot com a distribuição das notas para cada raça
- Terceira pergunta:
  - regiao-notas.csv: tabela que contém a relação de região e nota para todos os candidatos
  - regiao-notas.png: gráfico box-plot com a distribuição das notas para cada região

- Quarta pergunta:
  - renda-regiao-norte.csv: tabela que contém os dados agregados de cada grupo de renda para a região Norte
  - renda-regiao-nordeste.csv: tabela que contém os dados agregados de cada grupo de renda para a região Nordeste
  - renda-regiao-sudeste.csv: tabela que contém os dados agregados de cada grupo de renda para a região Sudeste
  - renda-regiao-sul.csv: tabela que contém os dados agregados de cada grupo de renda para a região Sul
  - renda-regiao-centro-oeste.csv: tabela que contém os dados agregados de cada grupo de renda para a região Centro Oeste
  - renda-regiao.png: gráfico de barras empilhadas com a porcentagem dos grupos de renda para cada região
- Quinta pergunta:
  - cota-regiao-norte.csv: tabela que contém os dados agregados de cada tipo de ação afirmativa para a região Norte
  - cota-regiao-nordeste.csv: tabela que contém os dados agregados de cada tipo de ação afirmativa para a região Nordeste
  - cota-regiao-sudeste.csv: tabela que contém os dados agregados de cada tipo de ação afirmativa para a região Sudeste
  - cota-regiao-sul.csv: tabela que contém os dados agregados de cada tipo de ação afirmativa para a região Sul
  - cota-regiao-centro-oeste.csv: tabela que contém os dados agregados de cada tipo de ação afirmativa para a região Centro Oeste
  - cota-regiao.png: gráfico de barras empilhadas com a porcentagem dos tipo de ação afirmativa para cada região

Todos os arquivos podem ser encontrados na pasta data analysis no Github.

Observação: algumas entradas dos dados estavam vazias, por exemplo a nota de alguns candidatos. Essas entradas foram ignoradas nas análises.

5 Sistema

O sistema utilizado foi o Windows 10 Pro 64bit, AMD Ryzen 5 3600, memória de 16GB, e Ubuntu instalado na máquina virtual usando o WSL2.

Softwares utilizados:

- Visual Studio Code
- WSL
- Excel Online
- draw.io

O relatório foi feito no Overleaf.

6 Modelos

O dois modelos utilizam código de cores para facilitar a visualização: as chaves primárias das dimensões possuem a cor correspondente da sua chave estrangeira na tabela fato.

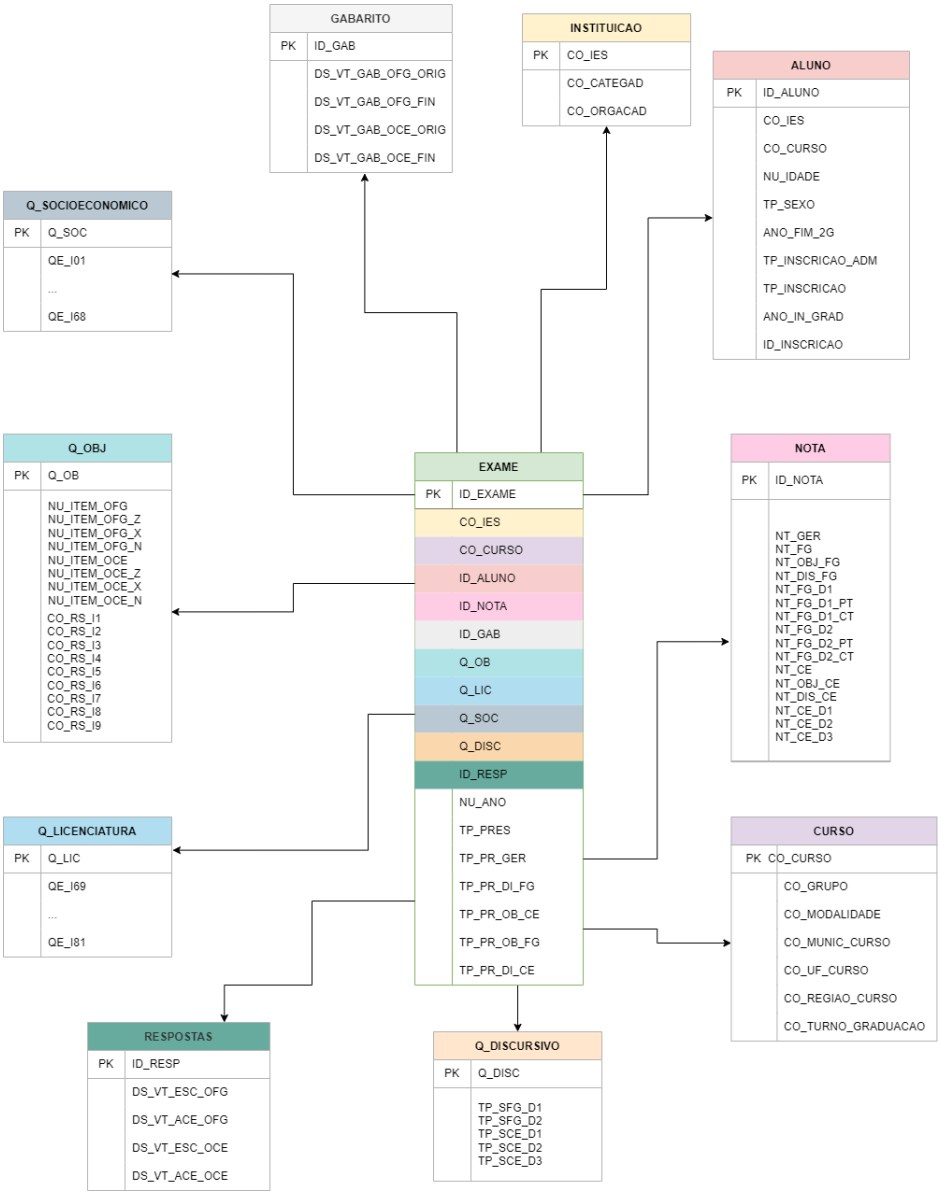

Figura 1: Primeiro modelo, antes das correções necessárias para a criação do banco

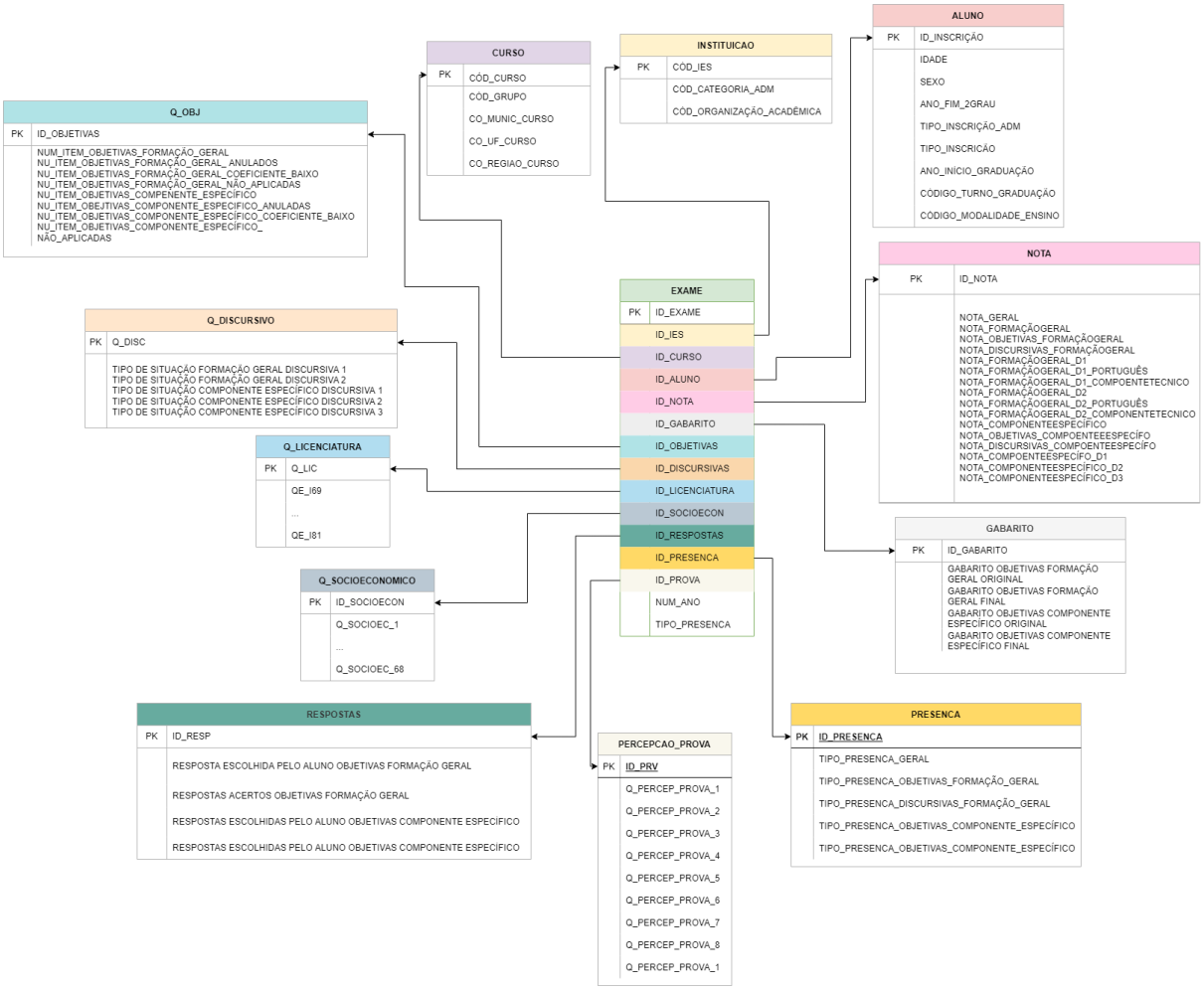

Figura 2: Segundo modelo, que foi utilizado para criar o banco, já corrigido.

7 Gráficos

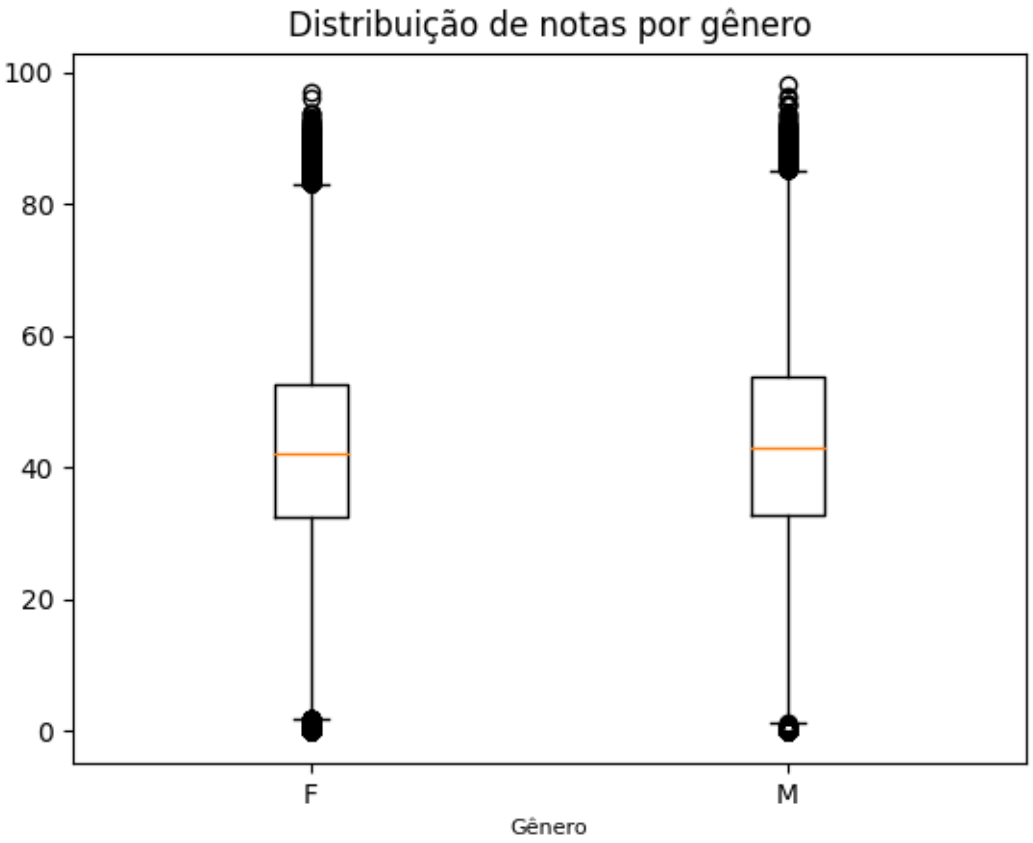

Figura 3: Distribuição de notas por gênero

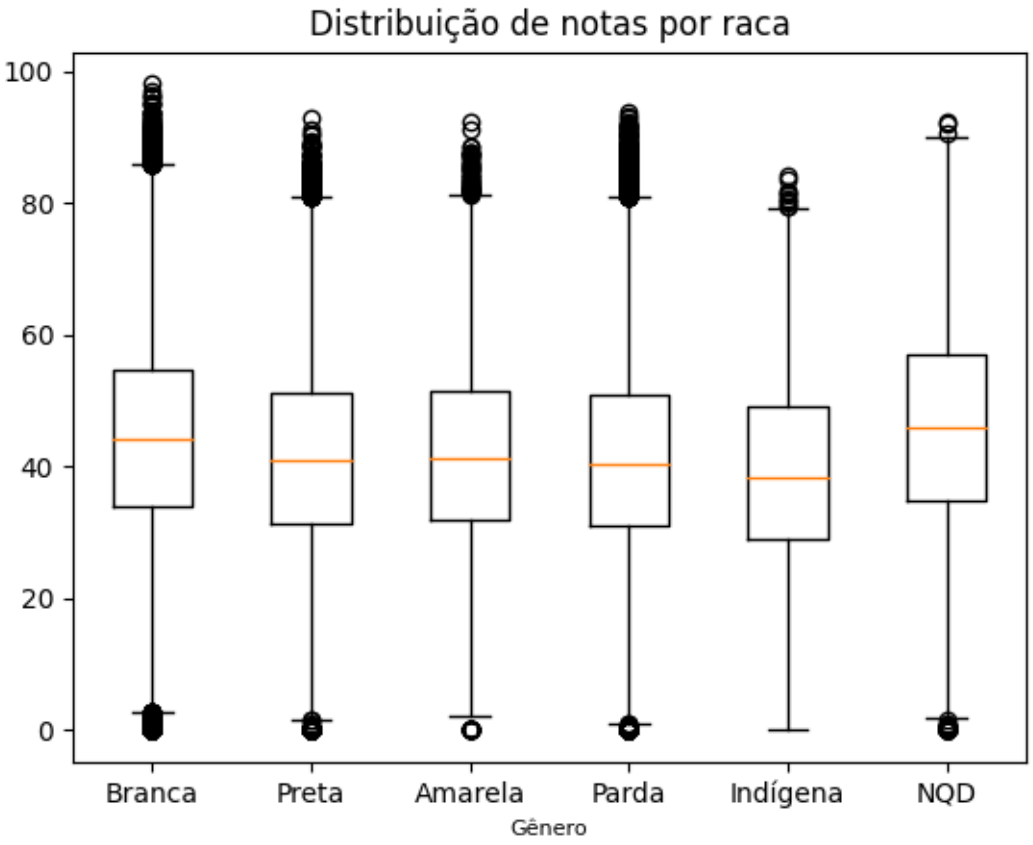

Figura 4: Distribuição de notas por raça

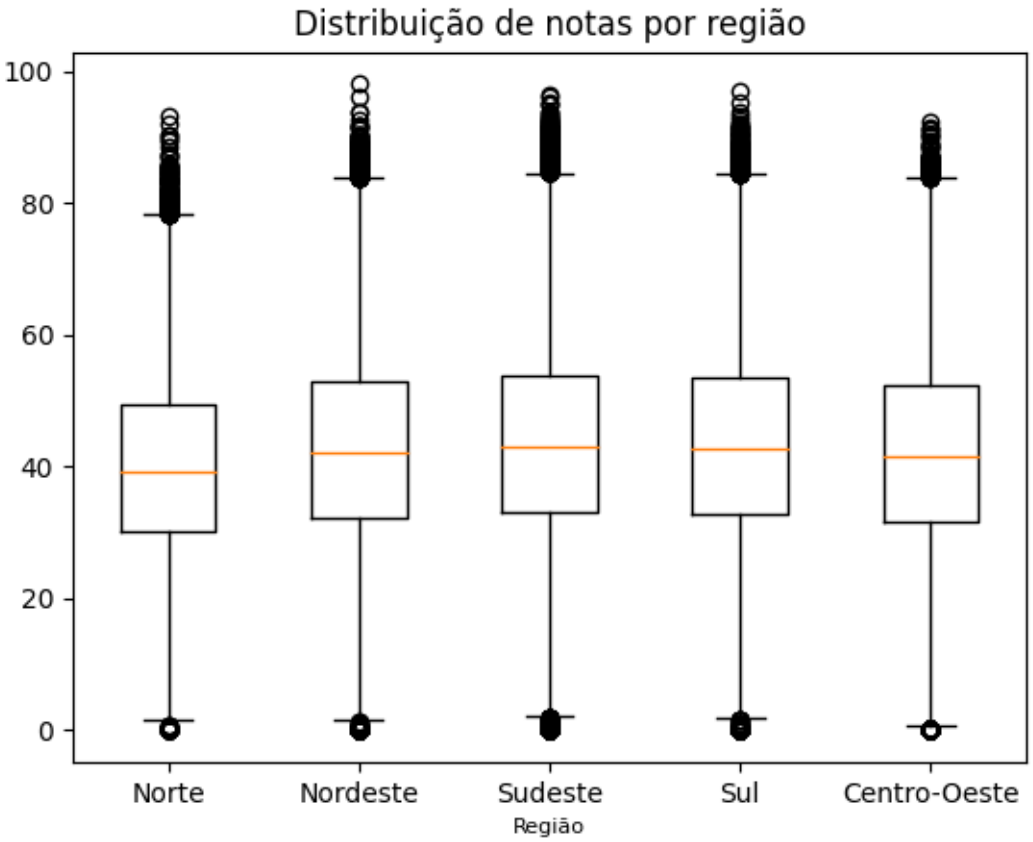

Figura 5: Distribuição de notas por região

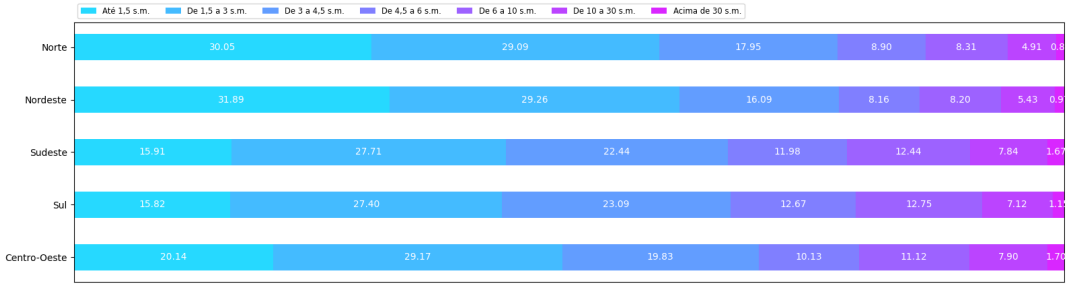

Figura 6: Renda média por região

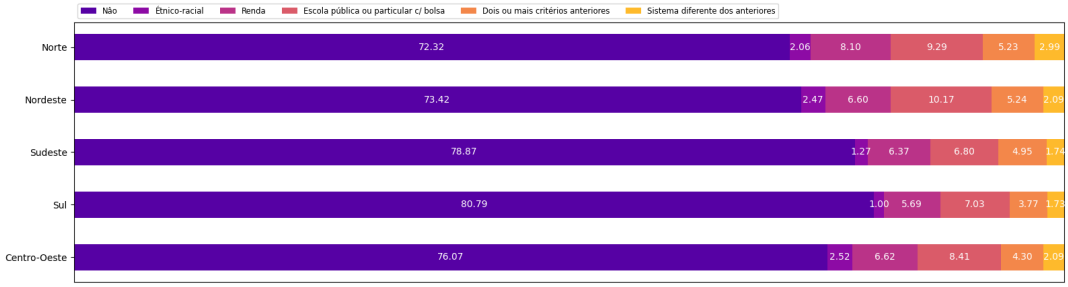

Figura 7: Políticas de ação afirmativa por região