UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATAS
ESTATÍSTICA

Bruno Henrique Rodrigues

Análise Causal Exploratória aplicada aos dados do Enem 2021:
revelando a estrutura causal dos fatores explicativos da nota final

Juiz de Fora
2023

Bruno Henrique Rodrigues

Análise Causal Exploratória aplicada aos dados do Enem 2021:
revelando a estrutura causal dos fatores explicativos da nota final

Trabalho de Conclusão de Curso apresentado ao curso de bacharelado em Estatística da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do título de bacharel em Estatística.

Orientador: Dr. Augusto Carvalho Souza
Coorientador: Dr. Marcel de Toledo Vieira

Juiz de Fora
2023

Ficha catalográfica elaborada através do programa de geração
automática da Biblioteca Universitária da UFJF,
com os dados fornecidos pelo(a) autor(a)

Rodrigues, Bruno Henrique.
Análise Causal Exploratória aplicada aos dados do Enem 2021 : revelando a estrutura causal dos fatores explicativos da nota final / Bruno Henrique Rodrigues. -- 2023.
88 p. : il.

Orientador: Augusto Carvalho Souza
Coorientador: Marcel de Toledo Vieira
Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Juiz de Fora, Instituto de Ciências Exatas, 2023.

1. Análise Causal Exploratória. 2. Inferência Causal. 3. Directed Acyclic Graphs. 4. Enem. 5. Desempenho escolar. I. Souza, Augusto Carvalho, orient. II. Vieira, Marcel de Toledo, coorient. III. Título.

Bruno Henrique Rodrigues

Análise Causal Exploratória aplicada aos dados do Enem 2021:
revelando a estrutura causal dos fatores explicativos da nota final

Trabalho de Conclusão de Curso apresentado ao curso de bacharelado em Estatística da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do título de bacharel em Estatística.

Aprovada em (dia) de janeiro de 2023

BANCA EXAMINADORA

_____

Dr. Augusto Carvalho Souza - Orientador
Universidade Federal de Juiz de Fora

_____

Dr. Marcel de Toledo Vieira - Coorientador
Universidade Federal de Juiz de Fora

_____

Dra. Ângela Mello Coelho
Universidade Federal de Juiz de Fora

_____

Dr. Ricardo da Silva Freguglia
Universidade Federal de Juiz de Fora

“Feliz o que pode conhecer as causas das coisas.”

– Virgílio

RESUMO

Este trabalho tem como objetivo apresentar a relação causal entre variáveis socioeconômicas com a nota média do Exame Nacional do Ensino Médio (Enem) e entre si. O Enem foi criado em 1998 como ferramenta para avaliar o desempenho do estudante após o ensino básico e vem sendo usado como método de entrada no ensino superior desde 2009. Estudos anteriores mostraram que a renda familiar e a escolaridade dos pais são influentes no desempenho escolar do estudante. Quanto à investigação das relações entre as variáveis, a Análise Causal Exploratória, ou Descoberta Causal, é o processo de inferir modelos causais, ou seja, a relação de causa-efeito das variáveis em um conjunto de dados, um complemento da inferência causal. A Inferência Causal, por sua vez, busca identificar o nível de causa de uma variável no efeito de outra. As relações de causa-efeito são apresentadas por grafos acíclicos direcionados (*Directed Acyclic Graphs*, DAGs), nos quais as relações são representadas por setas, que apontam o sentido do efeito e causa. Por meio de algoritmos baseados em restrição e baseados em escore, aplicados na base de dados dos participantes do Enem 2021, propomos DAGs que identificam fatores que podem explicar a nota média do Enem. Após manipulação no conjunto de dados, foram selecionadas as variáveis mais associadas ao desempenho escolar: tipo de escola, classe do Critério Brasil, escolaridade dos pais, sexo, cor/raça, acesso à internet em casa, localidade da escola e região do IBGE. Os testes de diferença de média utilizados retornaram diferença significativa entre as notas médias por grupo das variáveis selecionadas, mas pelo teste de verossimilhança para independência condicional feitos nos algoritmos baseados em restrição, as variáveis sexo do participante e localidade da escola não foram relacionadas com as demais variáveis, e acesso à internet apenas com a região do IBGE. Para os três algoritmos utilizados, os modelos propuseram influências diretas do tipo de escola e classe do Critério Brasil na nota do Enem, essa última sendo efeito, ou mediadora, das demais variáveis selecionadas. Após a remoção das variáveis sexo, localização da escola e acesso à internet em casa, a escolaridade da mãe também tem efeito direto na nota média do participante, junto com a classe do Critério Brasil e o tipo de escola, resultados que estão de acordo com trabalhos anteriores.

Palavras-chave: Análise Causal Exploratória; Inferência Causal; *Directed Acyclic Graphs*; Enem; Desempenho escolar.

ABSTRACT

This work aims to present the causal relation between socioeconomic variables and the mean grade on the Exame Nacional do Ensino Médio (Enem). The Enem was created in 1998 as a tool to measure the performance of students after concluding basic education, and has been used as an entry method for higher education since 2009. Previous studies showed that family income and the parents' education level are influential on the performance of the student. As for the investigation on the relation between variables, the Exploratory Causal Analysis (ECA), or causal discovery, is the process of inferring a causal model, that is, the cause-effect relation of the variables on a dataset, a complement for Causal inference. Causal Inference, on the other hand, aims to identify the cause level of a variable as the effect on another. The cause-effect relationship is presented as Directed Acyclic Graphs (DAGs), where those relations are represented as arrows, pointing the cause-effect direction. Using constraint-based and score-based algorithms for causal discovery, applied to the Enem 2021 dataset, we proposed DAGs that identified factors which can explain the students' average grade on Enem. After dataset manipulation, we select the variables most associated with school performance: type of school, income class, parental education, sex, color/race, access to the internet at home, location of school, and macroregion. The tests for difference between means returned significant differences between the groups of the selected variables, but testing conditional independence using likelihood tests for the constraint-based algorithms, sex, and location of school were not related to the other variables, and access to the internet at home being only related to the macroregion. For the three algorithms used, the models proposed direct influence of the type of school and income class on the Enem grade, the latter being influenced by, or mediating, the other variables. After removing variables sex, location of school, and access to the internet at home, the mother's education also had a direct effect on the student's Enem grade, along with income class and type of school, results that agree with previous studies.

Keywords: Exploratory Causal Inference; Causal Inference; Directed Acyclic Graphs; Enem; School performance.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

O Exame Nacional do Ensino Médio (Enem) foi implantado em 1998, com objetivo de avaliar o desempenho dos alunos após o fim do ensino básico. Desde 2009, o Enem é usado como mecanismo de acesso para o ensino superior. As notas do Enem podem ser utilizadas no Sistema de Seleção Unificada (Sisu) e no Programa Universidade para Todos (ProUni), além dos participantes poderem, também, pleitear financiamento estudantil em programas do governo, como o Fundo de Financiamento Estudantil (Fies) (INEP, 2022)**.**

Variáveis já explicadas em trabalhos anteriores, como a renda média familiar, a escolaridade dos pais, majoritariamente da mãe, essa sendo mais representativa que a renda, são alguns dos fatores significativos no desempenho escolar do aluno, resultados estes encontrados tanto em literatura nacional quanto internacional (MENEZES-FILHO, 2007; BARROS *et al*., 2001; HANUSHEK, 1989). No entanto, outras possíveis variáveis influentes da nota final no Enem bem como a estrutura de relação entre elas podem ser encontradas através da Análise Causal Exploratória.

A Análise Causal Exploratória, ou Descoberta Causal, é o processo de inferir o modelo causal, ou seja, a relação de causa-efeito entre variáveis, por meio de métodos estatísticos em um conjunto de dados (SPIRTS *et al*., 2012; TALEBI, 2021). Os principais métodos para a análise são os algoritmos baseados em restrição, que constroem o modelo causal testando independência condicional, e algoritmos baseados em pontuação, que gera diferentes candidatos para o modelo e define o melhor testando pontuações entre os modelos candidatos (SHEN *et al*., 2020). Realizada a Descoberta Causal, podemos seguir com a Inferência Causal, que busca identificar o grau da causa no efeito (ZHANG, 2008).

A Inferência Causal pode ser descrita como a definição e análise de efeitos em um tratamento (NEYMAN, 1990), ou como o processo de determinar o quanto uma ligação em uma observação reflete uma relação de causa e efeito (JACOB & GANGULI, 2016). As principais abordagens atualmente para inferência causal, por sua vez, são duas: os “Resultados Potenciais" e as “Equações Estruturais Causais”. Um modelo de equações estruturais causais pode ser visto como uma combinação de análise fatorial e regressão, ou a ampliação destas para a análise de trajetórias e caminhos (BOMTEMPO, 2005). Tal técnica é amplamente utilizada em áreas como Ciências Sociais (MORGAN & WINSHIP, 2015) e Epidemiologia (SILVA, 2021). Como

exemplo, temos uma das primeiras tentativas de formulação dessa combinação pelo geneticista Sewall Wright (1921) que utilizou equações e grafos para apresentar uma relação causal entre o sintoma e a doença. Neste exemplo de Wright, tomando X como a variável “doença” e Y como um sintoma qualquer da doença, podemos definir uma equação linear:

$$y = \beta x + u_Y \quad (1)$$

Nessa equação, x é a gravidade da doença, y é a gravidade do sintoma e u_Y é qualquer outro fator que possa afetar o sintoma Y quando X for constante. Ao interpretar essa equação, devemos pensar que a natureza, antes de definir o valor de Y, “consulta” os valores de x e u e com com base nesses valores define o valor para a variável Y com base em $y = \beta x + u_Y$. De forma similar, podemos definir a equação $x = u_X$ para explicar a ocorrência da doença X, onde U_X representa todos valores desconhecidos que afetam X (PEARL, 2010).

No entanto, a equação (1) ainda não expressa a relação causal do processo, porque equações algébricas são objetos simétricos. Reescrevendo a equação (1) como (2), podemos pensar equivocadamente que o sintoma é causa da doença:

$$x = (y - u_Y)/\beta \quad (2)$$

Como forma de expressar o direcionamento por trás desse processo, Wright ampliou a equação com um diagrama, posteriormente chamado de “diagrama de caminho”, no qual setas são desenhadas das causas para os efeitos observados. (WRIGHT, 1921; PEARL, 2010).

Para se ter uma visão mais clara sobre a Inferência Causal a partir da abordagem das Equações Estruturais Causais, fazemos uso dos Grafos Direcionados Acíclicos (DAGs, do inglês *Directed Acyclic Graph*). Os DAGs são conjuntos de vértices/nodos (variáveis) e arestas/setas/vetores (efeitos causais) que irão conectar os vértices um ao outro (SCHEINES, 1996). Para esses grafos, um efeito partindo de uma variável não pode terminar nela mesma (MORGAN & WINSHIP, 2015). DAGs causais trazem uma representação parcimoniosa dos aspectos qualitativos do processo gerador de dados, com letras (ex.: X, Y, Z) representando as variáveis aleatórias e setas representando as possíveis relações causais (PEARL, 2010;

GLYMOUR & JEWELL, 2016). Por exemplo, $X \rightarrow Y$ representa a possível relação causal de X em Y.

Temos três importantes e não exclusivas formas de associações/estruturas que formam um conjunto de DAG, e as respectivas relações de (in)dependência, que podem ser abertas ou fechadas/bloqueadas (CINELLI *et al*., 2022):

- **Cadeia/mediador** é um padrão representado por $X \rightarrow Y \rightarrow Z$, em que a variável X tem um efeito causal no desfecho Z, tendo Y como mediadora. Se condicionarmos no mediador Y, todo o efeito causal de X em Z desaparece, pois o único caminho causal de X para Z passa por Y. Esse condicionamento em Y bloqueia o fluxo de associação causal entre X e Z, que flui pela "porta da frente" e faz sumir a correlação entre X em Z. Assim, a associação incondicional identifica, sem viés, o efeito causal de X em Z;
- **Garfo/causa comum** indica um viés de confundimento. O padrão $X \leftarrow Y \rightarrow Z$ indica que X e Z compartilham uma causa comum Y, induzindo uma associação não-causal entre ambas variáveis. Condicionar Y em uma estrutura de garfo bloqueia o fluxo de associação, fazendo com que X e Z exibam uma independência condicional, isto é: $X \perp\!\!\!\perp Z \mid Y$;
- **Garfo invertido** representa efeito comum. Para $X \rightarrow Y \leftarrow Z$, Y é um efeito comum de X e Z. Diferente das outras duas estruturas, um efeito comum, por padrão, não induz associação entre X e Z. Porém, o condicionamento em Y induz uma associação não causal entre as duas outras variáveis.

Com isso, por meio de DAGs, podemos expressar a equação (1) como $U_X \rightarrow X \rightarrow Y \leftarrow U_Y$ (a ligação de X para Y sendo o parâmetro β). A ausência de um vetor de Y para X representa a afirmação que o sintoma Y não está entre os fatores U_X que afetam a doença X (PEARL, 2010). As variáveis U_X e U_Y, por sua vez, indicam fatores desconhecidos, observados ou não, que são intencionalmente omitidos no modelo. Essas variáveis desconhecidas, externas em relação ao modelo observado, são chamadas de "exógenas", ou independentes, enquanto as conhecidas, internas em relação ao modelo observado, são chamadas de "endógenas", ou dependentes (PEARL, 2010; BOMTEMPO, 2005).

Para as variáveis X, Y, Z, se X e Y não têm conexão com Z, como por cadeia ou causa comum, então X e Y estão direcionalmente separadas, ou d-separadas, por Z,

e caso qualquer conexão com Z, então X e Y estão direcionalmente conectadas, ou d-conectadas, por Z (KALISCH *et al*., 2012; SILVA, 2021). Esta análise de d-separação permite identificar as estruturas dos DAG em relação às dependências e independências, condicionais ou marginais, e, com isso, identificar DAGs com diferentes estruturas, mas que produzem as mesmas relações de dependência e independência entre as variáveis. Isto é, se dois DAGs A e B, com o mesmo conjunto de variáveis, possuem os mesmos d-separadores, então podemos dizer que existe uma equivalência de Markov entre A e B (ZHANG, 2008).

Ainda, para estudarmos causalidade sem viés, o mesmo indivíduo deveria ser submetido a diversas situações ou tratamentos, consequentemente tendo vários desfechos ou resultados potenciais (NEYMAN, 1990). Sendo assim, na Inferência Causal é necessário que o pesquisador reduza a realidade, geralmente isolando o efeito de uma única causa ou tratamento em um único desfecho (SILVA, 2021).

Utilizando um conjunto de dados disponibilizado pelo Inep, buscamos gerar modelos que representem o efeito do tipo de escola, pública (estadual ou municipal) ou privada, e outras variáveis socioeconômicas no desempenho do Exame Nacional do Ensino Médio (Enem) do ano de 2021, através da Descoberta Causal. Na seção de Metodologia, o conjunto de dados do Enem e as ferramentas para a análise e Descoberta Causal são apresentados. Na seção de Resultados, a seleção das variáveis é justificada pela Análise Exploratória dos dados e Testes de Diferença de Média entre os grupos de variáveis, e apresentados os diagramas dos fatores explicativos da nota média no Enem propostos pelos algoritmos de descoberta causal. Por fim, são discutidos estes resultados e como eles estão de acordo com trabalhos de anos anteriores.

2 METODOLOGIA

2.1 DADOS

O trabalho aqui apresentado foi realizado utilizando os dados públicos do Enem de 2021. O conjunto de dados original possui 3.389.832 registros e 76 variáveis, divididas entre: características do participante, da escola, do local de aplicação da prova, da prova objetiva, da redação e do questionário socioeconômico. As variáveis a seguir foram criadas para este trabalho:

- CRITERIO, para substituir o uso da renda média familiar e incluir a classe do poder de compra da família do participante, ou classe de renda, seguindo o Critério Brasil pela soma de pontos de cada variável (listadas no Anexo A):
  - Classe A: 45-100 pontos;
  - Classe B1: 38-44 pontos;
  - Classe B2: 29-37 pontos;
  - Classe C1: 23-28 pontos;
  - Classe C2: 17-22 pontos;
  - Classe DE: 0-16 pontos.
- REGIAO, para agrupar as unidades federativas das escolas por grande região do IBGE;
- NOTA_FINAL, para calcular a nota final pela média aritmética das competências da prova, como na equação (3):

$$Nota\ final\ = \frac{Português + Matemática + Ciências\ Humanas + Ciências\ Natureza + Redação}{5} \quad (3)$$

Buscando reduzir vieses na nota, os dados foram filtrados da seguinte forma:

- apenas os participantes presentes nos dois dias de prova;
- participantes de escolas em atividade;
- participantes que responderam ao tipo de escola (TP_ESCOLA) equivalente à dependência administrativa da escola (TP_DEPENDENCIA_ADM_ESC), ou seja, escolas municipais e estaduais com dependência pública e particulares com dependência particular (houveram casos como o tipo de

escola sendo particular com a dependência administrativa sendo estadual, e vice-versa);

- participantes que já concluíram o ensino médio ou iriam concluir naquele ano;
- participantes que não fizeram a prova apenas com intuito de treinar o conhecimento, ou seja, que não havia concluído e nem concluiria em 2021;
- participantes do ensino médio regular.

As escolas federais foram removidas pois, apesar de serem públicas, uma vez que as notas médias por competência estavam em equivalência com as notas das escolas privadas.

2.2 DESCOBERTA CAUSAL

Diferentes algoritmos com implementações no R foram utilizados para geração de modelos para identificação da estrutura de relações causais entre as variáveis analisadas e a nota final do aluno. Os tipos de algoritmos utilizados no trabalho foram baseados em *restrição*, que identifica independências condicionais por meio de testes estatísticos, e baseados em *escore*, nos quais os possíveis modelos de DAG são avaliados por uma pontuação.

2.2.1 R

As análises foram realizadas no ambiente R/RStudio (R CORE TEAM, 2022), com a programação em Anexo A. Os principais pacotes utilizados para realização do trabalho foram:

- `dplyr` (WICKHAM, 2009) para tratamento dos dados;
- `ggplot2` (WICKHAM, 2009) para geração de gráficos;
- `pcalg` (KALISCH *et al*., 2022) para os algoritmos baseados em restrição;
- `micd` (FORAITA & WITTE, 2021) para teste de independência condicional;
- `bnlearn` (SCUTARI *et al*., 2021) para o algoritmo baseado em pontuação.

2.2.2 ALGORITMO PC

O algoritmo PC, baseado em restrições, é um dos algoritmos mais antigos baseados em amostras independentes e identicamente distribuídas (iid), assumindo que não há confundidores latentes. Esse algoritmo oferece uma arquitetura de busca na qual vários procedimentos estatísticos para decisão de independência condicional são usados (SPIRTES *et al*., 2001; GLYMOUR *et al*., 2019).

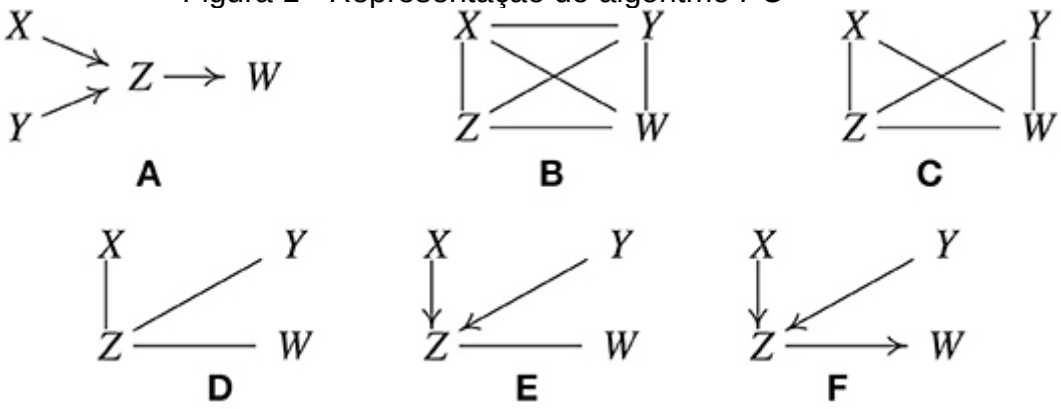
Para exemplificar o funcionamento do algoritmo PC, temos a estrutura real de um determinado DAG (figura 1A). Por d-separação, essa estrutura implica que X é independente de Y, ou seja, $X \perp\!\!\!\perp Y$, e tanto X quanto Y são independentes de W condicionando em Z, ou seja, $\{X, Y\} \perp\!\!\!\perp W \mid Z$. O algoritmo PC funciona da seguinte forma (GLYMOUR *et al*., 2019):

1. Forma-se um grafo conectando todas as variáveis, sem direções (Figura 1B);
2. Através dos testes de independência, elimina-se os vetores entre variáveis que são independentes incondicionalmente, sendo aqui o vetor entre $X - Y,$ pois $X \perp\!\!\!\perp Y$ (Figura 1C);
3. Para cada dupla de variáveis (A, B) conectadas e para cada variável C conectada com alguma dessas duas variáveis, elimina-se o vetor entre A e B se $A \perp\!\!\!\perp B \mid C$, então aqui são eliminados os vetores $X - W$ e $Y - W$, pois $X \perp\!\!\!\perp W \mid Z$ e $Y \perp\!\!\!\perp W \mid Z$ (Figura 1D);
4. Para cada dupla de variáveis (A, B) conectadas e para cada par de variáveis (C, D) conectadas, se as variáveis C e D estão conectadas a A ou a B, elimina-se o vetor entre A e B se $A \perp\!\!\!\perp B \mid \{C, D\}$;
5. Para cada tripla de variáveis (A, B, C), tais que A e B sejam adjacentes, B e C sejam adjacentes, mas A e C não sejam adjacentes, direciona-se os vetores nessa tripla como $A \rightarrow B \leftarrow C$ (garfo invertido), caso B não esteja no conjunto em que A e C sejam independentes fazendo com que o vetor entre elas seja eliminado, aqui sendo $X \rightarrow Z \leftarrow Y$ (figura 1E);
6. Para cada tripla de variáveis (A, B, C) tais que $A \rightarrow B - C$, A e C não sendo adjacentes, define-se a direção $A \rightarrow B \rightarrow C$ (cadeia), aqui sendo $Y \rightarrow Z \rightarrow W$ (Figura 1F);

7. Caso não seja possível encontrar a relação entre as variáveis, o vetor continuará sem direção.

Com esses passos, o modelo encontrado em 1F se iguala ao real, 1A.

Figura 1 - Representação do algoritmo PC



Fonte: Glymour *et al*. (2019)

Para este trabalho, foi utilizada a função `pc` do pacote `pcalg` do R, cujos parâmetros estão detalhados em Kalisch *et al*. (2022). Para os testes de independência condicional, foi usado o Teste de Razão de Verossimilhança, pela função `micd`, sendo atualmente o único compatível com o pacote `pcalg` para teste com variáveis tanto categóricas quanto numéricas.

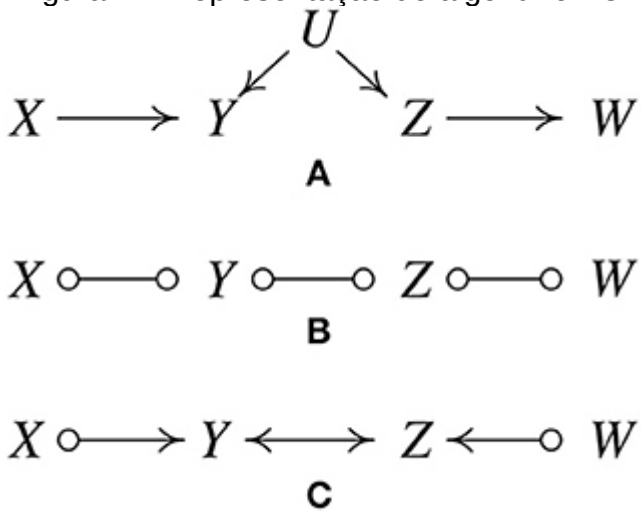
2.2.3 ALGORITMO FCI

O Algoritmo FCI (*Fast Causal Inference*) também é baseado em restrições, mas assumindo que possa haver confundidor oculto. Os passos iniciais do Algoritmo FCI são iguais aos do PC, variando no momento de definir os vetores entre os pares de nós. No FCI, os vetores são inicialmente representados por "o – o", onde cada "o" pode ser o início ou o fim do vetor. Com isso, apenas testar a independência entre as variáveis $X - Y$ não é o suficiente para remoção do vetor entre elas, dada a possibilidade de existirem variáveis ocultas. O vetor aberto entre $X\text{o} - \text{o}Y$ é testado para remoção em um próximo passo, calculando possíveis d-separadores para X e para Y, e testando independência de X e Y dado todos os possíveis subconjuntos dos possíveis d-separadores (GLYMOUR *et al*., 2019).

No exemplo da figura 2A, este sendo o modelo real, temos a variável não observada U sendo causa comum para Y e Z. Já passando do passo de remoção dos

vetores das variáveis independentes, seguimos com o modelo da figura 2B. Após os testes de possíveis d-separadores, a figura 2C apresenta um vetor bidirecional entre Y e Z, que indica pelo menos uma causa comum oculta entre as duas. Os "o" restantes em X e W representam que o algoritmo não foi capaz de identificar se há conexão direta entre $X - Y$ e entre $W - Z$.

Figura 2 – Representação do algoritmo FCI



Fonte: Glymour *et al*. (2019)

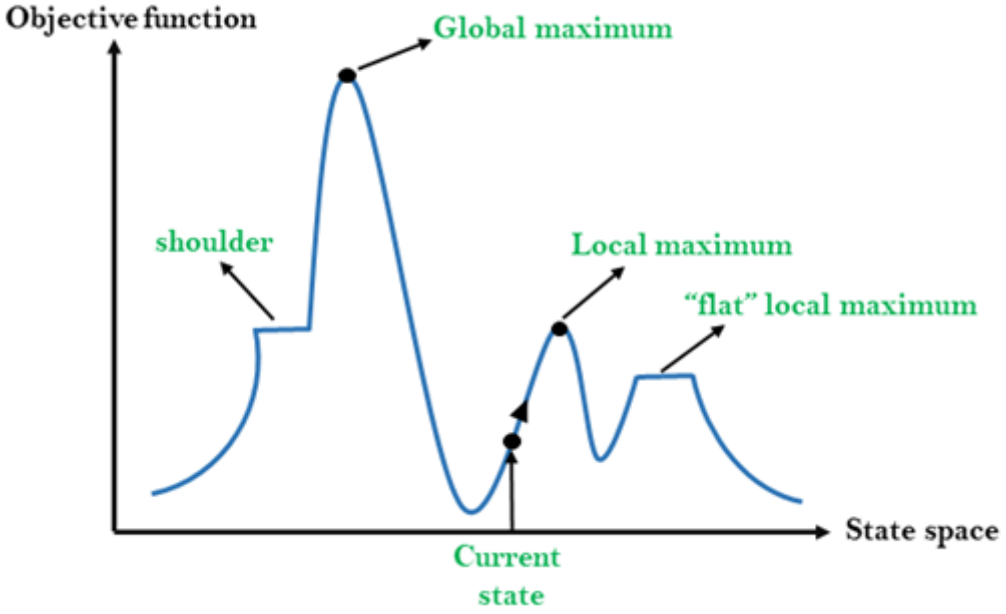
A implementação utilizada foi pela função `fci`, também do pacote `pcalg` no R, com parâmetros detalhados em Kalisch *et al*. (2022). Os parâmetros utilizados foram os mesmos da função `pc`, apenas excluindo o parâmetro `solve.confl`, pois não existe na função `fci`. Para os testes de independência condicional, também foi usado o Teste de Razão de Verossimilhança, pela função `micd`.

2.2.4 ALGORITMO *HILL-CLIMBING*

Para o algoritmo baseado em pontuação, foi utilizado o *hill-climbing* (escalada de montanha) que partindo de um ponto aleatório inicial, avalia qual ponto vizinho possui melhor resultado, ou seja, mais escala a montanha. No contexto de inferência causal, o algoritmo seleciona os pares de nós (A, B) e avalia as operações "adicionar relação", "trocar direção da relação", "remover relação" e seleciona o resultado que gera o melhor BIC (*Bayesian Information Criterion*) para o modelo (BERETTA *et al*. 2018; WANG, 2018). Uma limitação do algoritmo *hill-climbing* é que ele busca pelo máximo local, não necessariamente sendo o máximo global, então mesmo que seja encontrada uma boa solução, essa não é garantida de ser a melhor (Figura 3). A

implementação utilizada no R foi a partir da função `hc` do pacote `bnlearn`, com parâmetros detalhados em Scutari (2022).

Figura 3 – Representação do algoritmo *hill-climbing*



Fonte: Great Learning (2020)

2.2.5 TESTE DE DIFERENÇA DE MÉDIA

Para evidenciar as diferenças de médias nos grupos das variáveis, foi utilizado o Teste HSD de Tukey, que testa as comparações entre pares de médias de tratamento, $H_0: m_A = m_B$ x $H_1: m_A \neq m_B$. Sua expressão (4) é dada por (TUKEY, 1949):

$$dms = q\sqrt{\frac{QMR}{2}\left(\frac{1}{r_A} + \frac{1}{r_B}\right)} \quad (4), \text{ tais que}$$

- dms é a distância mínima significativa;
- q é o valor tabelado por Tukey em função do número de tratamento e dos graus de liberdade do resíduo;
- QMR é o Quadrado Médio do Resíduo da análise de variância;
- r_A e r_B são o número de repetições nos tratamentos A e B, respectivamente.

Se a diferença entre as médias for maior ou igual a distância mínima significativa, então as médias se diferem. No ambiente R, foi utilizada a função `TukeyHSD`.

2.2.6 APLICAÇÃO

Após refinamentos no conjunto de dados e descarte das variáveis do participante, como número de inscrição ou dados de identificação da escola, foram realizadas a Análise Exploratória dos dados e Testes de Diferença de Média utilizando o HSD de Tukey, e posteriormente os diagramas iniciais. As variáveis iniciais selecionadas foram:

- Tipo de escola (TP_ESCOLA, nos diagramas sendo ESC):
  - Escola pública;
  - Escola particular.
- Critério Brasil (CRITERIO, nos diagramas sendo CRIT):
  - A;
  - B1;
  - B2;
  - C1;
  - C2;
  - DE.
- Escolaridade do pai (Q001) e Escolaridade da mãe (Q002):
  - A - Nunca estudou;
  - B - Não completou a 4ª série/5º ano do Ensino Fundamental;
  - C - Completou a 4ª série/5º ano, mas não completou a 8ª série/9º ano do Ensino Fundamental;
  - D - Completou a 8ª série/9º ano do Ensino Fundamental, mas não completou o Ensino Médio;
  - E - Completou o Ensino Médio, mas não completou a Faculdade;
  - F - Completou a Faculdade, mas não completou a Pós-graduação;
  - G - Completou a Pós-graduação;
  - H - Não sei.

- Localização da escola (TP_LOCALIZACAO_ESC, nos diagramas sendo LOC):
    - Urbana;
    - Rural.
- Cor/raça do participante (TP_COR_RACA, nos diagramas sendo COR):
    - Branca;
    - Parda;
    - Preta;
    - Amarela;
    - Indígena;
    - Não declarada.
- Sexo do participante (TP_SEXO, nos diagramas sendo SEX):
    - Masculino;
    - Feminino.
- Acesso à internet em casa (Q025, nos diagramas sendo NET):
    - Sim;
    - Não.
- Região do IBGE da escola do participante (REGIAO, nos diagramas sendo REG):
    - Norte;
    - Nordeste;
    - Centro-Oeste;
    - Sudeste;
    - Sul.

O mesmo conjunto de dados foi utilizado nos três algoritmos apresentados. Para os algoritmos baseados em restrição, as ligações forçadas entre as variáveis e parâmetros (quando existentes nas duas funções) foram iguais nas funções `pc` e `fci`. Na aplicação do algoritmo *hill-climbing*, os modelos iniciais propostos sugeriam relações causais entre variáveis que não fazem sentido para nós, como sexo do aluno explicando a localidade da escola. Para contornar esse problema, conexões e restrições foram impostas ao algoritmo, o qual também permite forçar a direção das relações causais entre as variáveis.

As variáveis SEX, LOC e NET não foram explicativas para a variável resposta Nota, nem para as outras variáveis, sendo removidas para geração dos novos modelos. Os modelos propostos removendo Q001 (escolaridade do pai) foram idênticos aos propostos removendo Q002 (escolaridade da mãe).

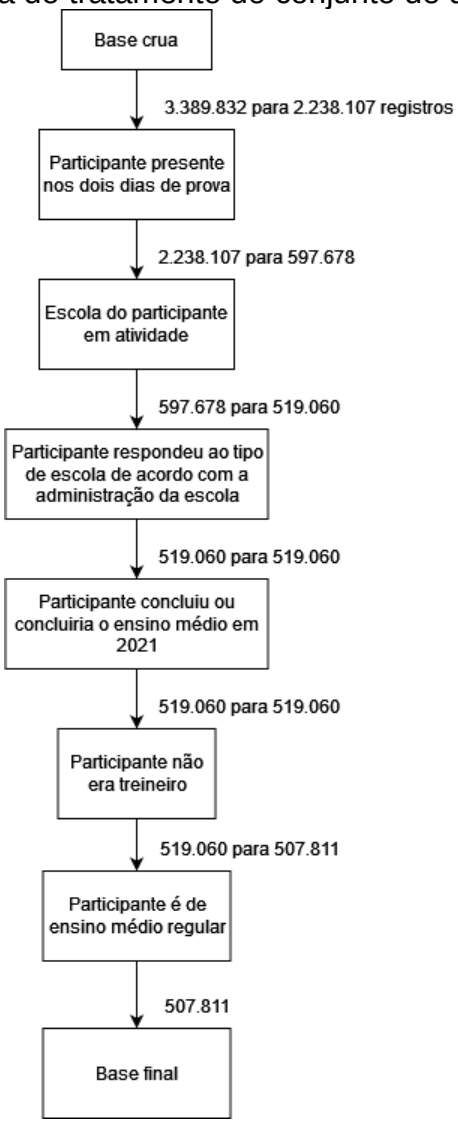
Todos os cálculos e gráficos foram gerados no ambiente RStudio (a programação segue no Apêndice B).

3 RESULTADOS

3.1 ANÁLISE EXPLORATÓRIA DE DADOS

Após os filtros, o conjunto de dados passou de 3.389.832 registros para 507.811 registros, aproximadamente 15% do total.

Figura 4 – Fluxograma do tratamento do conjunto de dados



Fonte: Elaborado pelo autor (2022)

Dividindo os participantes restantes por tipo de escola, aproximadamente 70% eram de escolas públicas e os 30% restantes de escolas particulares, como apresentado nas Tabelas 1a e 1b (as demais variáveis se encontram no Apêndice A):

Tabela 1a – Distribuição de alunos por administração da escola

Administração da escola	Freq.	%
Estadual	349.161	68,76%
Municipal	4.140	0,82%
Privada	154.510	30,42%
Total	**507.611**	**100,00%**

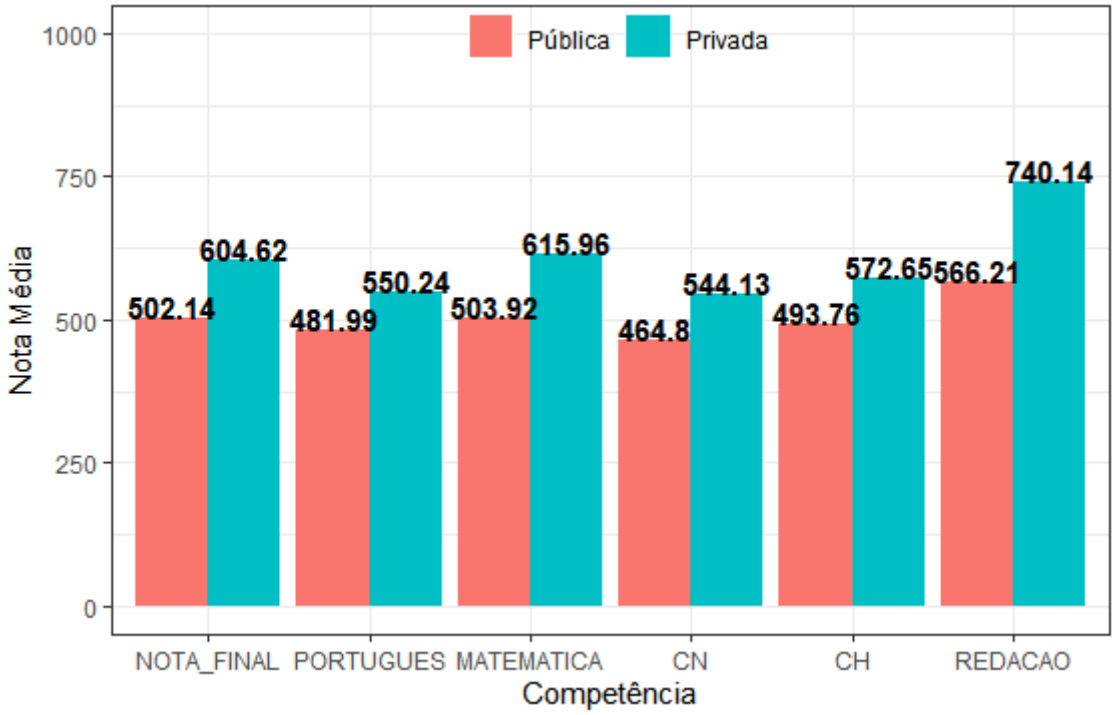
Tabela 1b – Distribuição de alunos por tipo de escola

Tipo de escola	Freq.	%
Pública	353.301	69,58%
Privada	154.510	30,42%
Total	**507.611**	**100,00%**

Fonte: Elaborado pelo autor (2022).

Analisando as notas da prova por competências e geral, os padrões foram os mesmos para os dois tipos de escola (Figura 5). Para nota média geral por tipo de escolas, as públicas ficaram com 502,1, enquanto as particulares ficaram com 604,6. A média nacional foi 533,3.

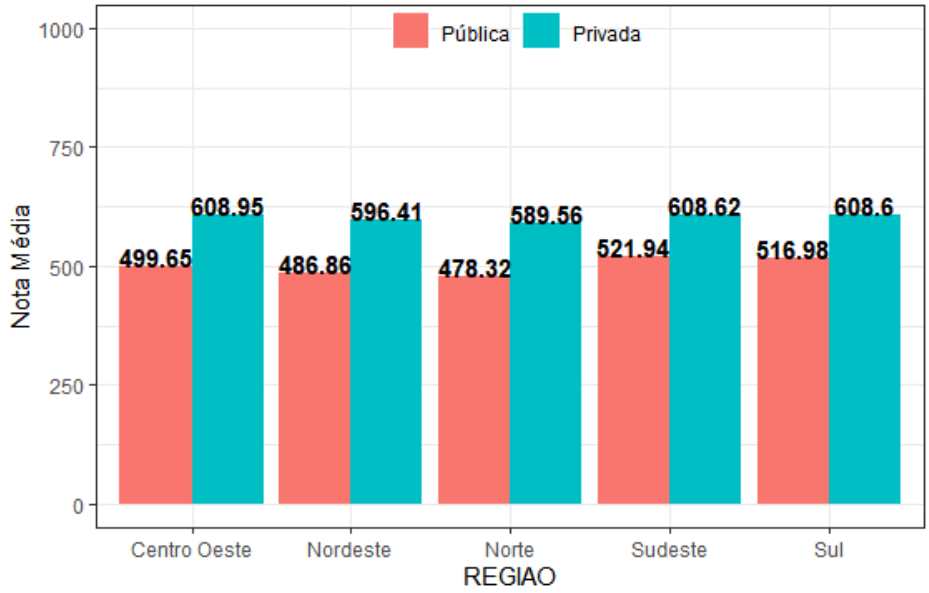
Figura 5 – Nota média por competência e tipo de administração da escola



Fonte: Elaborado pelo autor (2022)

A partir deste momento, as análises serão feitas apenas com as notas médias finais dos participantes. Quando comparamos o tipo de escola por região, os desempenhos de escolas públicas nas regiões Sul e Sudeste estavam acima das demais (Figura 6).

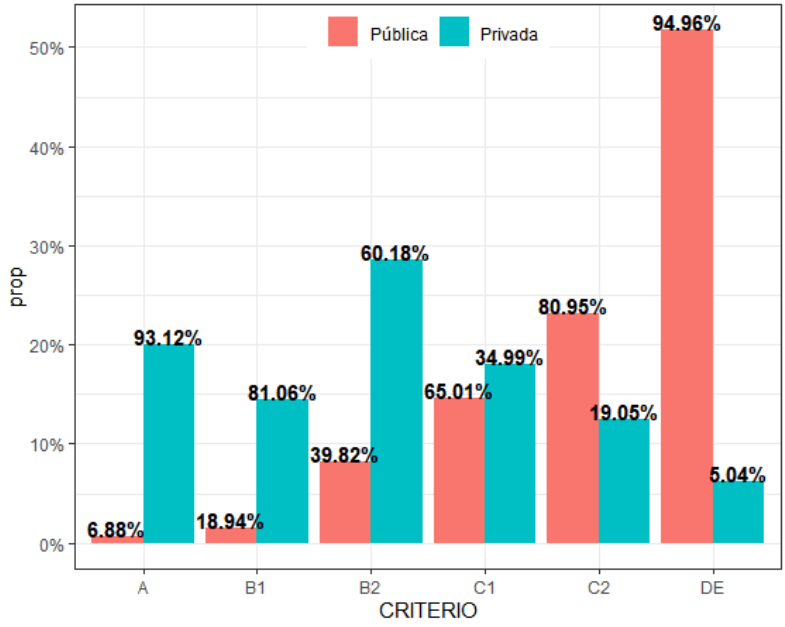
Figura 6 – Nota média geral por grande região e tipo de administração da escola



Fonte: Elaborado pelo autor (2022)

Na Figura 7, podemos ver que houve uma maior concentração de participantes de classe do Critério Brasil mais alta (esquerda do eixo x) em escolas particulares.

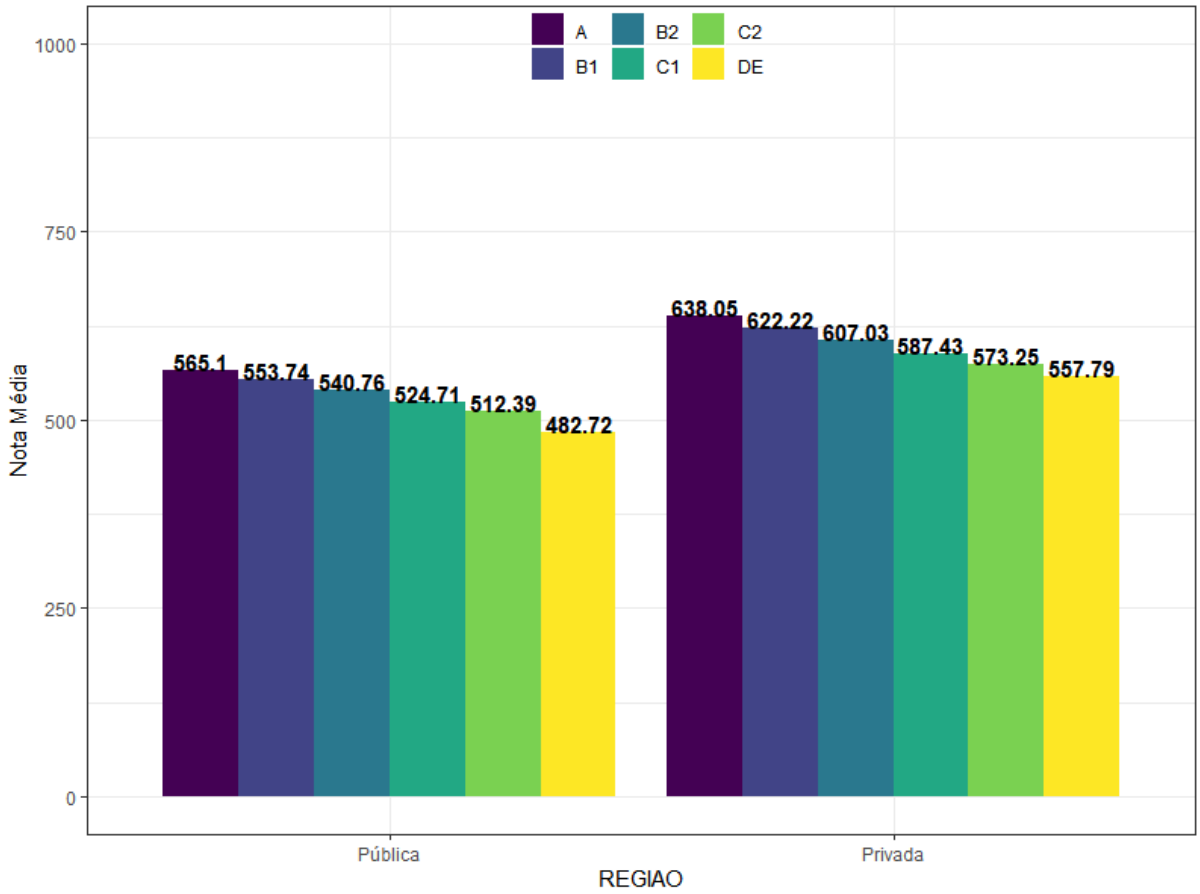
Figura 7 – Distribuição dos participantes por Critério Brasil



Fonte: Elaborado pelo autor (2022)

Também é perceptível que há uma correlação positiva entre a classificação no Critério Brasil e a nota média dos participantes, independentemente do tipo de escola e mesmo as classes mais baixas nas escolas particulares apresentam médias melhores que as classes mais altas nas escolas públicas (Figura 8).

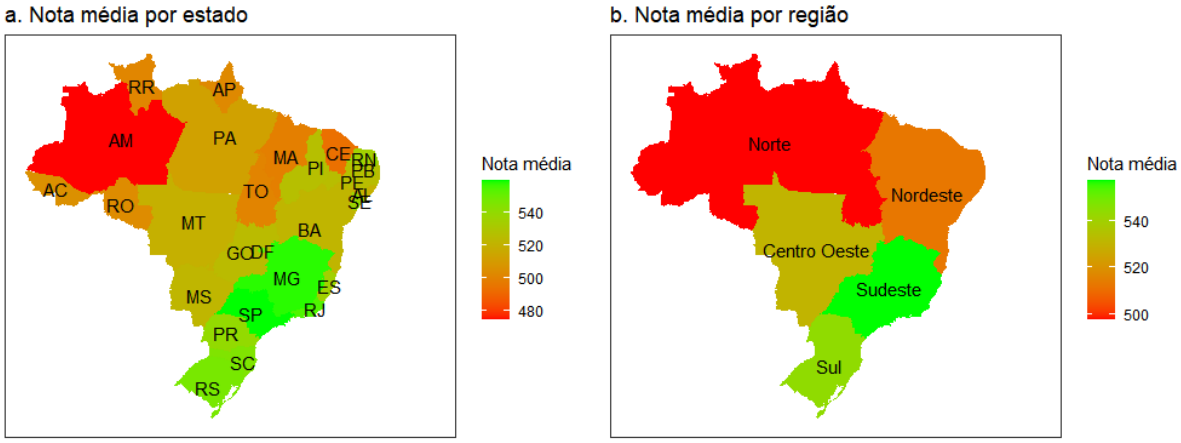
Figura 8 – Nota média por Critério Brasil no tipo de escola



Fonte: Elaborado pelo autor (2022)

Analisando as médias de notas por estado (Figura 9a), as três maiores médias foram na região Sudeste, sendo em ordem decrescente São Paulo, Minas Gerais e Rio de Janeiro. Em seguida, temos o Distrito Federal e os estados do Sul. Apesar da Região Nordeste ter apresentado a segunda menor média, o estado de Sergipe teve a 8ª maior média com 537,42. Para as grandes regiões (Figura 9b), as notas médias foram: 557,11 para o Sudeste; 543,11 para o Sul; 530,15 para o Centro-Oeste; 512,72 para o Nordeste; 497,62 para o Norte.

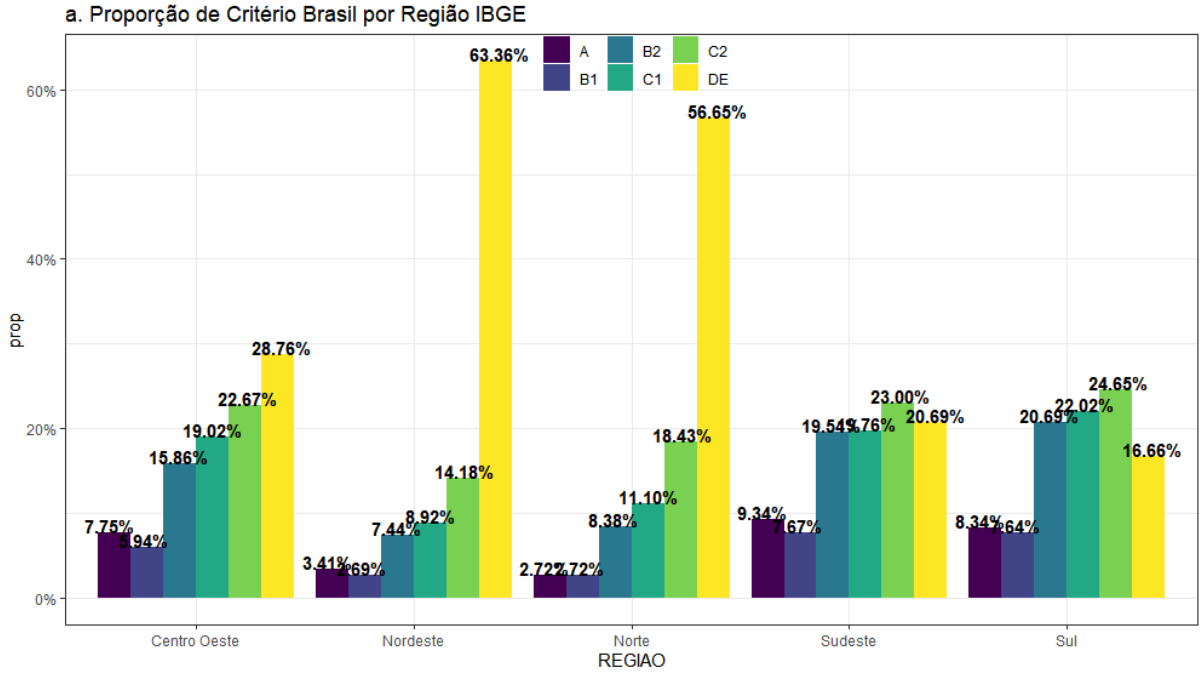
Figura 9 – Nota média por estado e região IBGE

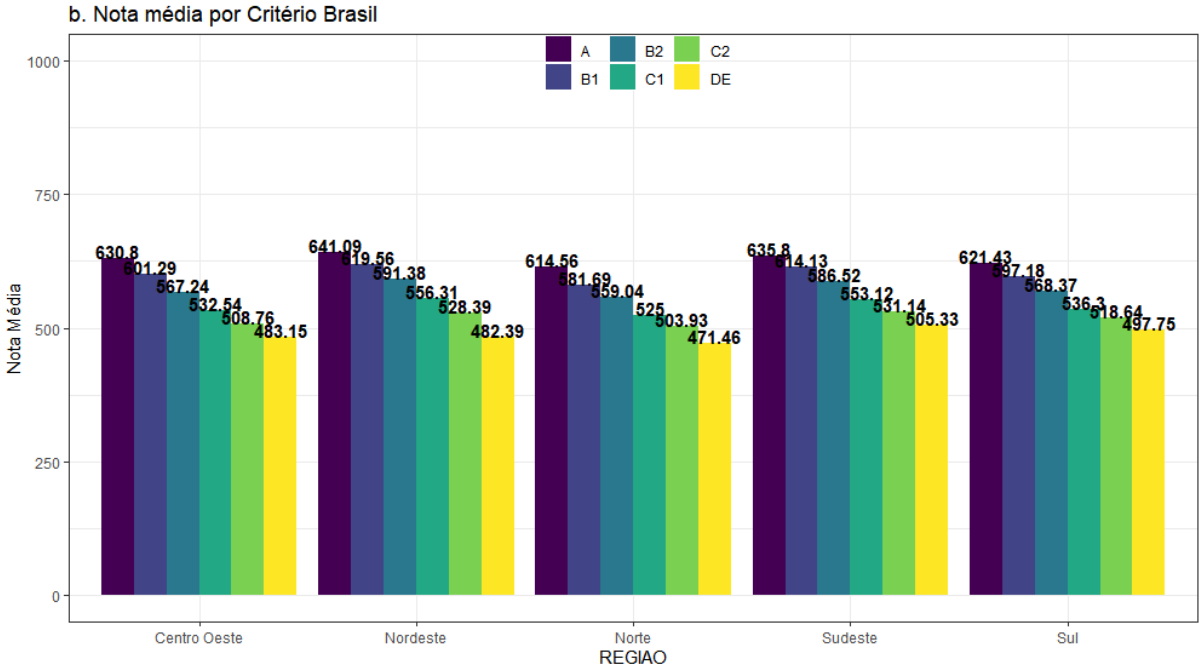


Fonte: Elaborado pelo autor (2022)

Pela distribuição das classes do Critério Brasil por região (Figura 10), temos grande disparidade nas regiões Nordeste e Norte para as classes mais baixas, as mesmas que apresentaram menores notas médias. 54,25% dos participantes da Classe DE estão na região Nordeste, e mesmo que no Norte sejam apenas 12,8%, menor que na região Sudeste, a classe DE foi a mais presente na região.

Figura 10 – Distribuições do Critério Brasil e nota média por grande região IBGE





Fonte: Elaborado pelo autor (2022)

Descartando as variáveis de inscrição para a Descoberta Causal, tivemos ainda oito variáveis para gerar modelos que possam explicar o processo gerador da nota do participante do Enem, todas categóricas: tipo de escola, sexo, cor/raça, localização da escola, região geográfica, escolaridade do pai, escolaridade da mãe, acesso à internet em casa. Através de testes de diferença de média, todas as variáveis apresentaram diferença significativa na nota média.

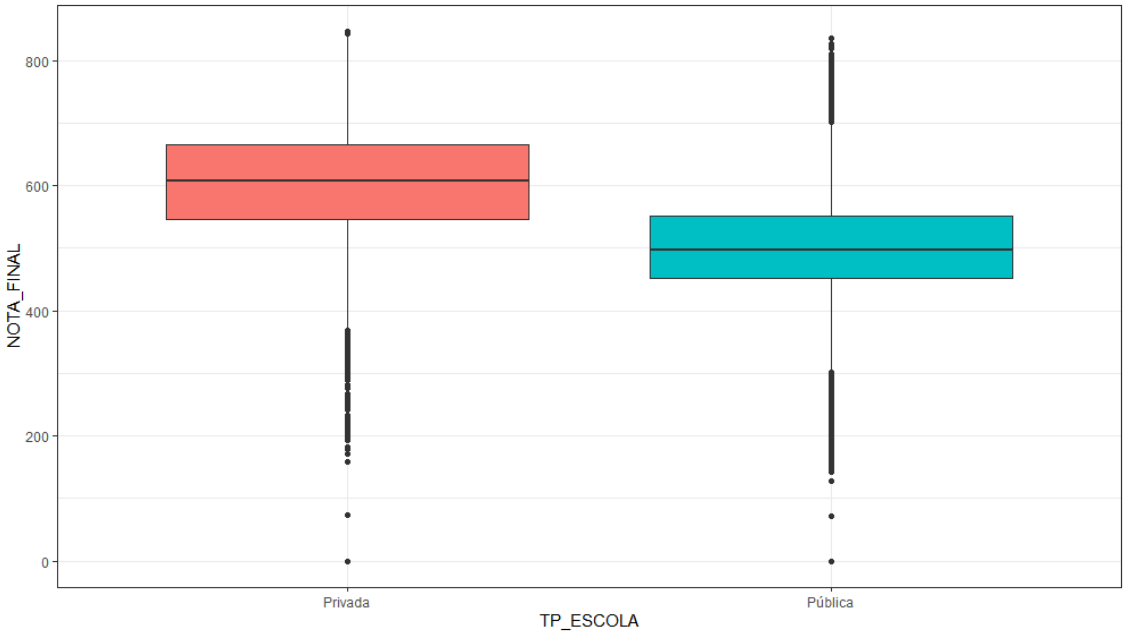
Durante os testes com os algoritmos baseados em restrições, as variáveis sexo do estudante e localidade da escola não apresentaram às outras variáveis, ou seja, não explicavam as outras variáveis nos modelos sugeridos, sendo removidas, portanto, dos testes posteriores. Foram também fixadas as ligações em classe do Critério Brasil e tipo de escola (CRIT–ESC), escolaridade do pai e nota do aluno (Q001–NOTA), e escolaridade da mãe e nota do aluno (Q002–NOTA).

3.2 TESTES DE MÉDIA

Ao realizarmos o teste HSD de Tukey nas variáveis selecionadas, com exceção de um caso na escolaridade da mãe (mostrado mais adiante), todos retornaram p-valor ≈ 0. Ou seja, há evidências de que as notas são significativamente diferentes entre as categorias das variáveis.

No boxplot da Figura 11, temos o primeiro quartil das notas das escolas particulares pouco abaixo do terceiro quartil das escolas públicas. Também nas escolas públicas, muitas notas estão acima do limite superior, próximas da nota média máxima das escolas particulares.

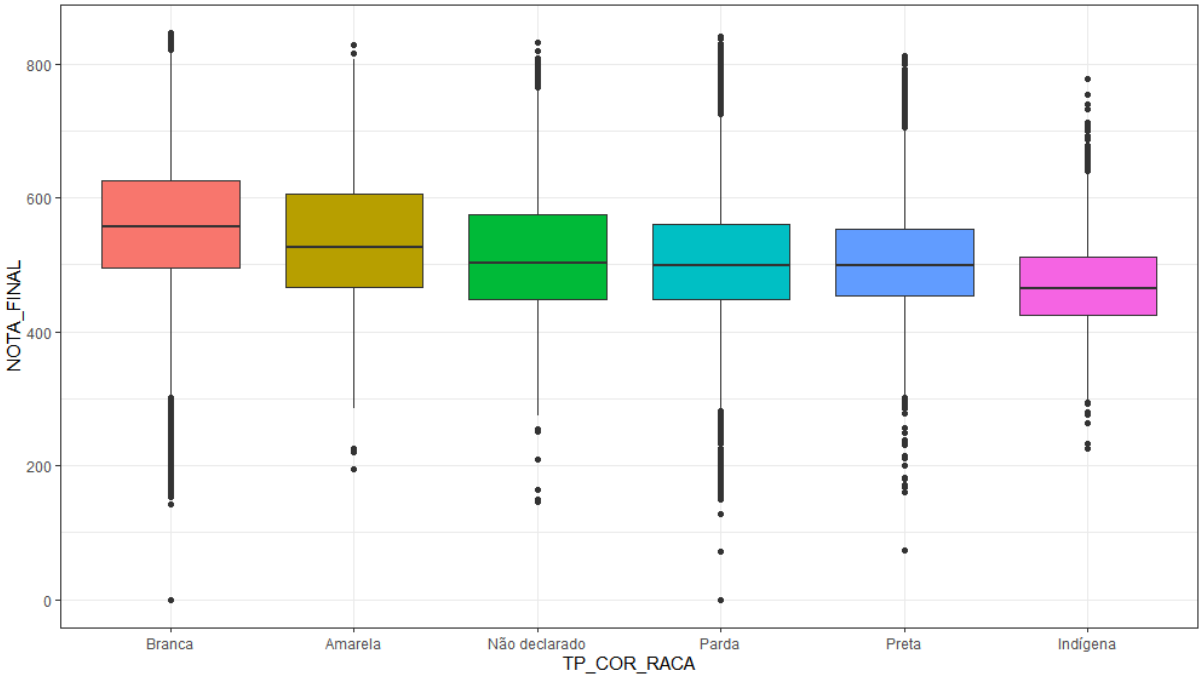
Figura 11 – Boxplot nota por tipo de escola



Fonte: Elaborado pelo autor (2022)

Na diferença de nota por cor, a comparação “Pardo” – “Preto” retornou p-valor aproximado 0,009, o único diferente de aproximadamente 0, mas, ainda significativo. A cor/raça indígena apresentou menores média, mediana e valor máximo, mas as notas estão menos dispersas e seu menor valor é maior que de outras cores/raças (Figura 12). Agrupando as cores/raças não brancas e realizando os mesmos testes, o p-valor ainda foi significativo.

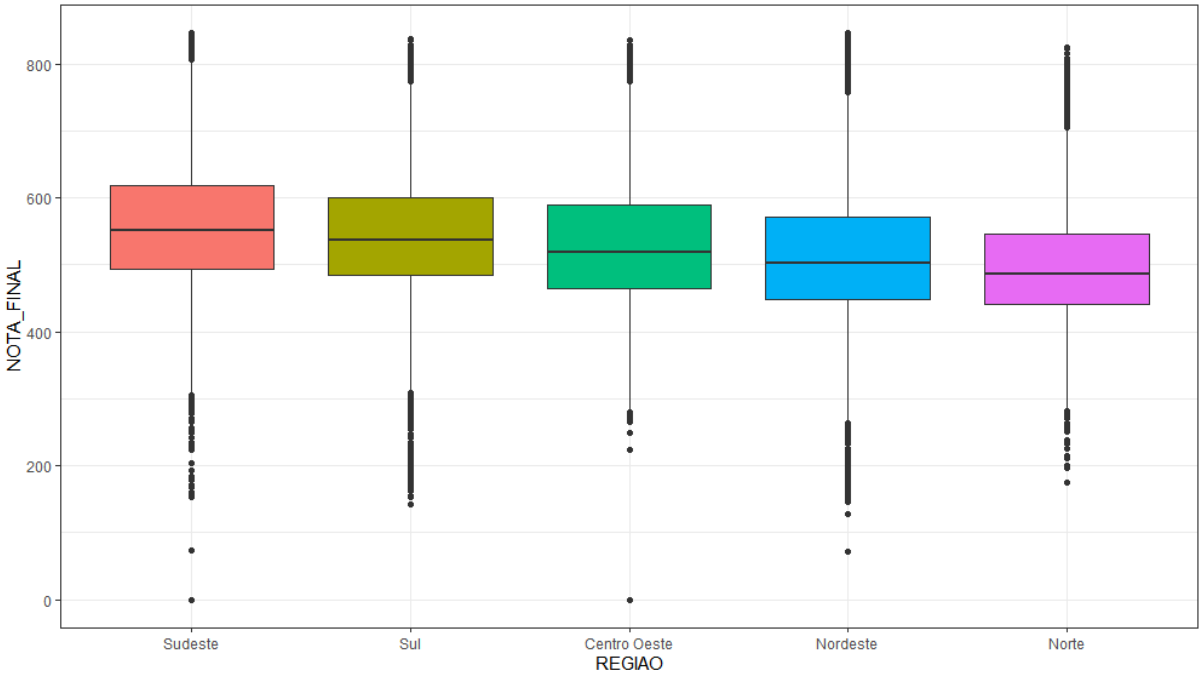
Figura 12 – Boxplot nota por cor/raça



Fonte: Elaborado pelo autor (2022)

Nas notas por região do IBGE (Figura 13), os pontos discrepantes superiores estavam próximos para todas as regiões.

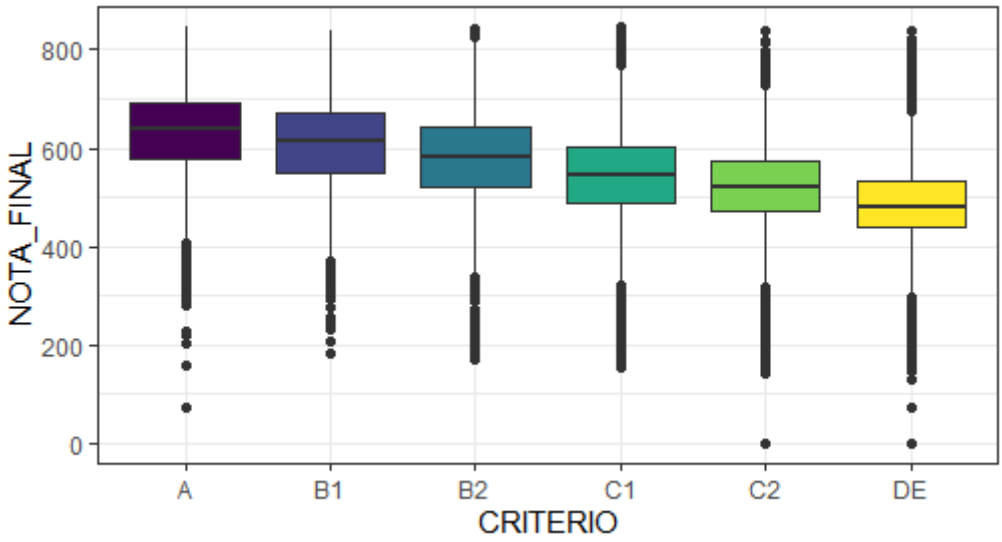
Figura 13 – Boxplot nota por região



Fonte: Elaborado pelo autor (2022)

Apesar de não ter aparecido nenhuma nota acima do limite superior para a Classe A do Critério Brasil, muitas apareceram abaixo do limite inferior, tendo notas até menores que das três classes abaixo dela (B1, B2, C1) (Figura 14).

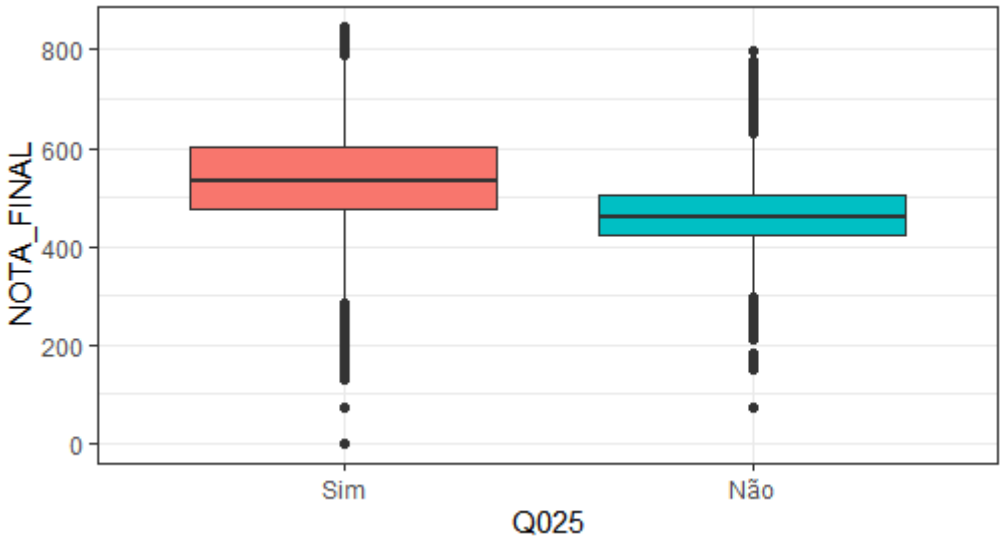
Figura 14 – Boxplot nota por Critério Brasil



Fonte: Elaborado pelo autor (2022)

Comparando as notas por acesso à internet (Figura 15), as notas médias dos que possuem acesso em casa foi mais dispersa do que a outra resposta. Mesmo com essa diferença significativa, a diferença de média entre as categorias não foi significativa para os testes de verossimilhança na Descoberta Causal, sendo removida da análise final.

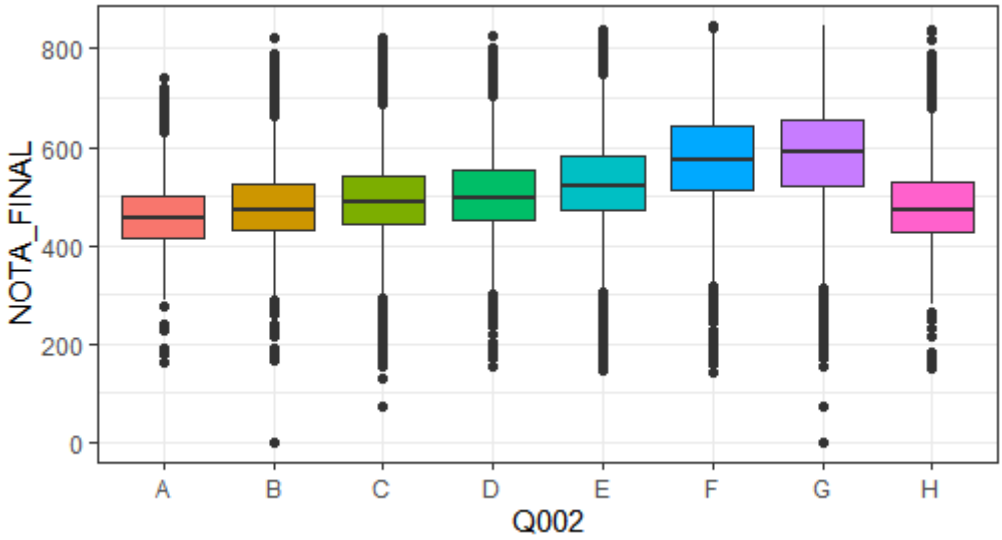
Figura 15 – Boxplot nota por acesso à internet em casa (Q025)



Fonte: Elaborado pelo autor (2022)

Os diferentes níveis de escolaridade da mãe também apresentaram diferença significativa na nota do participante, exceto no comparativo “Fundamental 1 incompleto”–“Desconhecido” (classes B–H), com p-valor de aproximadamente 0,105. Os boxplots desses dois grupos também são bem próximos (Figura 16).

Figura 16 – Boxplot nota por escolaridade da mãe (Q002)



Fonte: Elaborado pelo autor (2022)

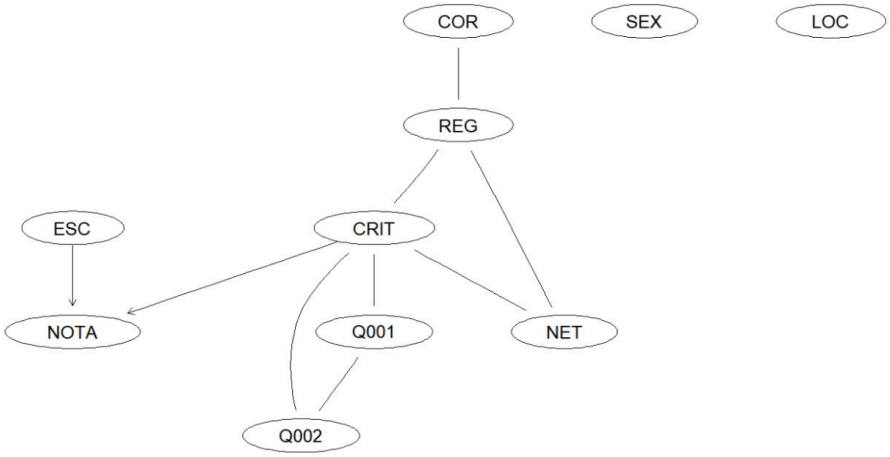
3.3 DESCOBERTA CAUSAL

As variáveis SEX, LOC e NET não foram explicativas para a variável resposta Nota, nem para as outras variáveis, sendo removidas para geração dos novos modelos. Os modelos propostos removendo Q001 (escolaridade do pai) foram idênticos aos propostos removendo Q002 (escolaridade da mãe).

3.3.1 ALGORITMO PC

Para o algoritmo PC, com as variáveis iniciais listadas em 2.2.5, o modelo propôs que o Critério Brasil e o tipo de escola explicam a nota, mas como seguem a estrutura de garfo invertido, não há associação causal entre as duas (Figura 17).

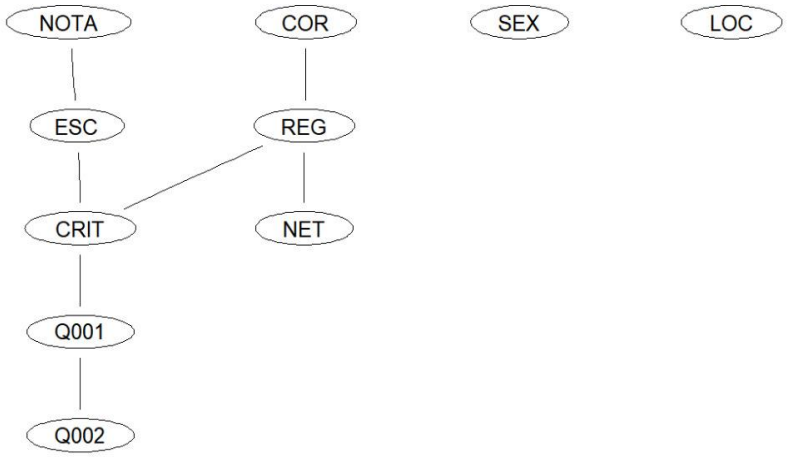
Figura 17 – Modelo proposto pela função `pc`



Fonte: Elaborado pelo autor (2022)

Ao fixar a relação Critério Brasil–Tipo de escola, sem direção, o algoritmo PC indicou ligação sem direção entre a nota e o tipo de escola, e a ligação entre nota e o Critério Brasil foi através do tipo de escola. Nenhuma das ligações apresentadas teve direção definida (Figura 18).

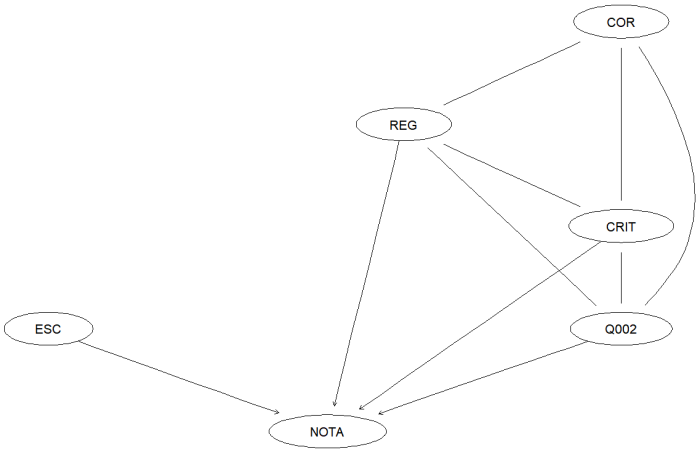
Figura 18 – Modelo proposto pela função `pc`, forçando a ligação CRIT–ESC



Fonte: Elaborado pelo autor (2022)

No modelo com a remoção das variáveis SEX, LOC, NET (Figura 19), o tipo de escola não teve relação direta com outras variáveis além da nota do participante, e a cor/raça do participante só teve relação com a nota através das outras variáveis. Diferente dos modelos anteriores, com mais variáveis, a região e a escolaridade da mãe tiveram também influência direta na nota do participante não só através do Critério Brasil. O mesmo modelo foi gerado ao forçar a ligação CRIT–ESC (sem direção definida), Q002–Nota, ou as duas ligações em conjunto.

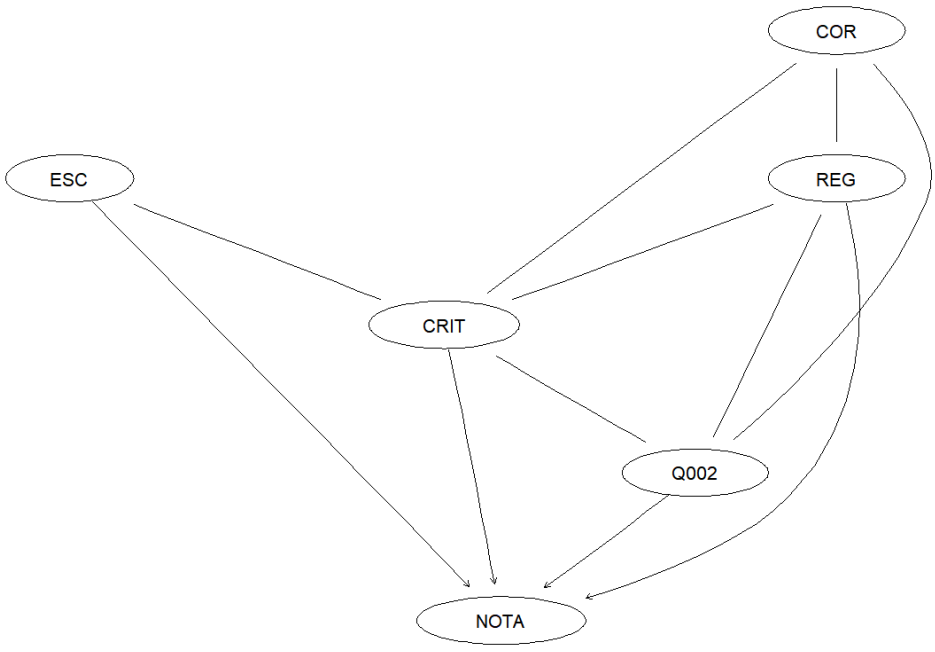
Figura 19 – Novo modelo proposto pela função `pc`



Fonte: Elaborado pelo autor (2022)

Além de forçar a ligação CRIT–ESC, também forçando tanto Q001–NOTA quanto Q002–NOTA, a nota foi explicada pelas duas escolaridades e também pelo tipo de escola. As escolaridades dos pais estavam relacionadas entre si, sem direção, mas apenas a do pai teve ligação direta com a classe do Critério Brasil, também sem direção. Novamente, o mesmo modelo foi gerado pelo algoritmo FCI mais adiante, com Critério Brasil explicando o tipo de escola, mas com variáveis não observadas entre as ligações (Figura 20).

Figura 20 – Novo modelo proposto pela função `pc`, forçando ligações

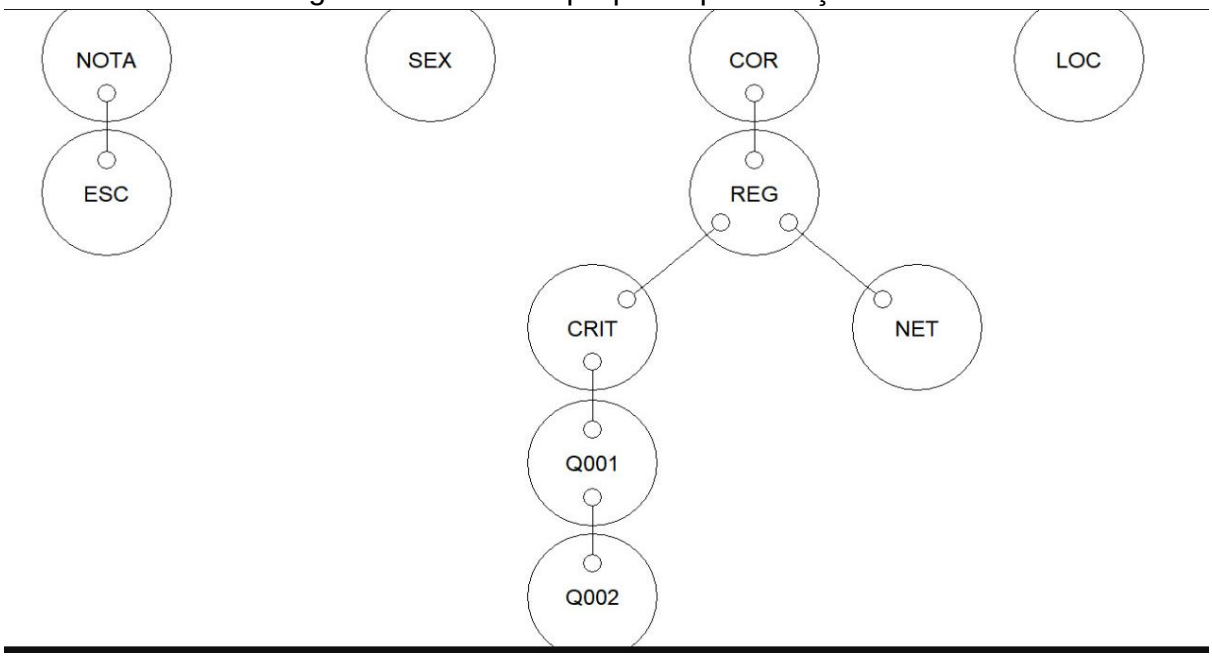


Fonte: Elaborado pelo autor (2022)

3.3.2 ALGORITMO FCI

O algoritmo FCI propôs um modelo inicial (Figura 21) que não foi equivalente ao gerado pelo algoritmo PC inicialmente. Neste modelo, apenas o tipo de escola teve relação com a nota, e com possível variável oculta confundidora. Novamente, sexo do participante ou localidade da escola não apresentou relação com as outras variáveis e as demais estavam ligadas ao Critério Brasil. Em alguns gráficos, as pontas das setas estão com sentido trocado ou até mesmo fora no meio da reta, sendo apenas um problema visual da função geradora do gráfico, não afetando o resultado apresentado.

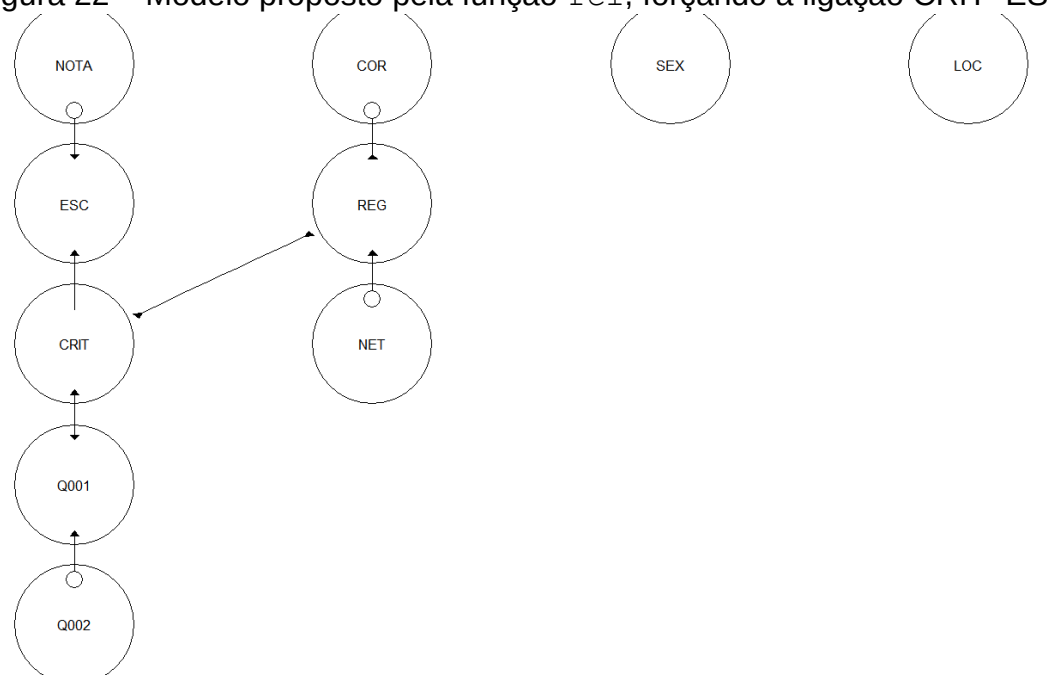
Figura 21 – Modelo proposto pela função `fci`



Fonte: Elaborado pelo autor (2022)

Ao forçar a ligação CRIT-ESC, o novo modelo (Figura 22) agora foi equivalente ao proposto pelo algoritmo PC. Aqui, o Critério Brasil explicou diretamente o tipo de escola e existiram variáveis não observadas entre Critério Brasil e região, Critério Brasil e escolaridade do pai; nas demais ligações, houve a possibilidade de também existirem variáveis ocultas.

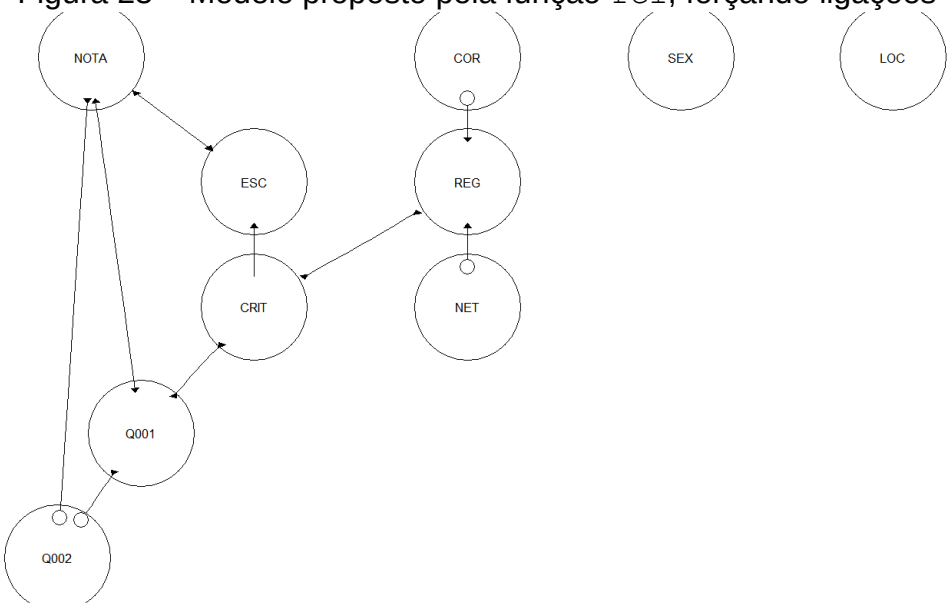
Figura 22 – Modelo proposto pela função `fci`, forçando a ligação CRIT–ESC



Fonte: Elaborado pelo autor (2022)

Ao fixar, simultaneamente, CRIT–ESC, Q001–NOTA e Q002–NOTA, o modelo proposto (Figura 23) também ficou equivalente ao proposto pela função `pc`, mas aqui houve uma relação direta do Critério Brasil no tipo de escola, e uma variável não observada confundidora entre o tipo de escola e a nota.

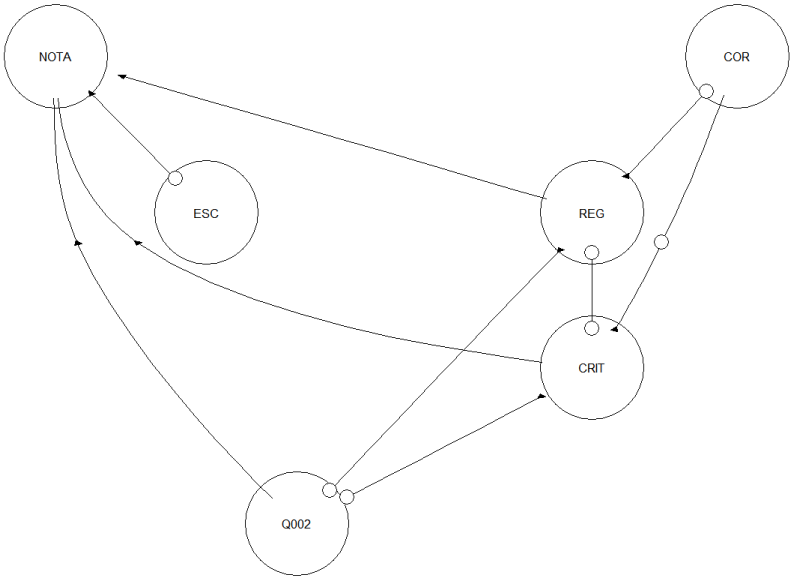
Figura 23 – Modelo proposto pela função `fci`, forçando ligações



Fonte: Elaborado pelo autor (2022)

Excluindo as variáveis SEX, LOC e NET, a função `fci` gerou um modelo em que o Critério Brasil explica diretamente o tipo de escola (Figura 24), ligação que não teve a direção definida pela função `pc`.

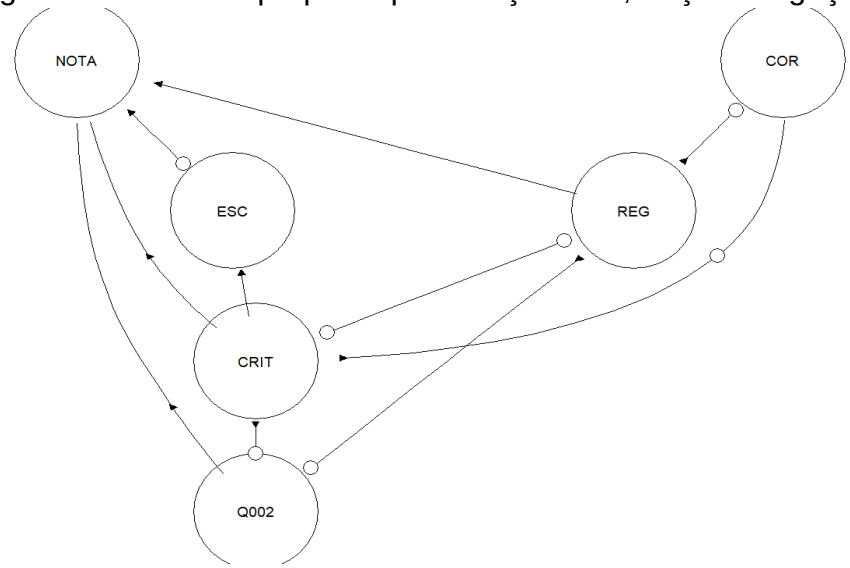
Figura 24 – Novo modelo proposto pela função `fci`[1]



Fonte: Elaborado pelo autor (2022)

Ao forçar a ligação entre CRIT–ESC, a única diferença do modelo anterior foi que agora o Critério Brasil explica diretamente o tipo de escola (Figura 25). Ao forçar também a relação Q002–NOTA, o mesmo modelo foi gerado.

Figura 25 – Modelo proposto pela função `fci`, forçando ligações



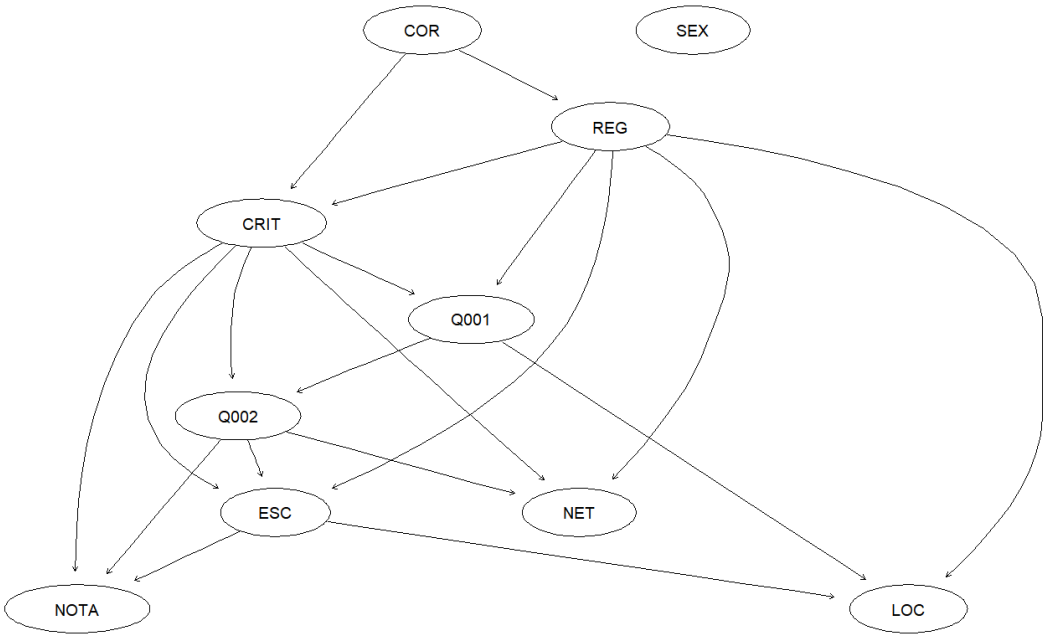
Fonte: Elaborado pelo autor (2022)

3.3.3 ALGORITMO *HILL-CLIMBING*

Na aplicação do algoritmo *hill-climbing*, os modelos iniciais propostos sugeriam relações causais entre variáveis que não fazem sentido para nós, como sexo do aluno explicando a localidade da escola. Para contornar esse problema, conexões e restrições foram impostas ao algoritmo, o qual também permite forçar a direção das relações causais entre as variáveis.

O tipo de escola foi explicado pelo Critério Brasil, e ambas explicavam a nota do participante. Com exceção da localidade da escola, o Critério Brasil explicou todas as demais variáveis, até o sexo do participante, tendo sido necessário forçar proibir tais direções das relações (Figura 26). O algoritmo propôs que a escolaridade da mãe, tipo de escola e o Critério Brasil explicavam a nota diretamente, além do Critério Brasil ser mediadora da cor/raça do participante e da região. A variável sexo não teve relação com nenhuma outra variável, e a localização da escola foi explicada, mas não explicou nenhuma outra variável.

Figura 26 – Modelo proposto pela função `hc`, após limitar ligações

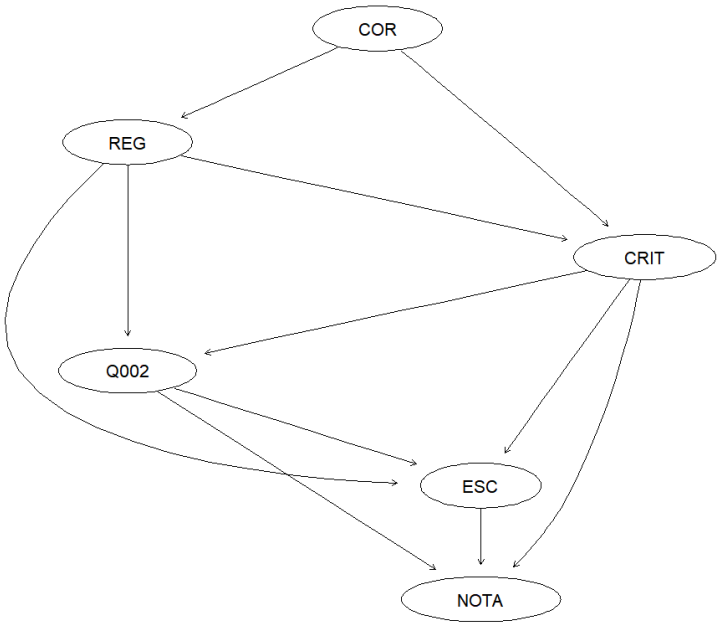


Fonte: Elaborado pelo autor (2022)

Removendo as mesmas variáveis listadas para os outros algoritmos, não foi necessário limitar as direções entre as variáveis restantes. Para este modelo, CRIT e Q002 foram confundidoras para o tipo de escola. Essas três variáveis continuaram

explicando a nota do participante diretamente, sem forçar ligação ou direção para a nota (Figura 27).

Figura 27 – Novo Modelo proposto pelo *hill-climbing*



Fonte: Elaborado pelo autor (2022)

4 DISCUSSÃO

A Análise Exploratória apresentou correlações entre a nota média e outras variáveis do conjunto de dados, como a classe do Critério Brasil, tanto para escolas públicas quanto escolas particulares. Os DAGs propostos auxiliam na geração de modelos que pretendem explicar numericamente o nível de impacto dessas variáveis selecionadas. Comparando as notas médias por região e tipo de escola, foi possível ver que as notas de escolas públicas, ainda que menores que das particulares, eram maiores nas regiões com maiores proporções das classes altas do Critério Brasil. A classe por si só não pode ser o único fator para o desempenho, como apresentado no boxplot de nota por classe (Figura 15) que tiveram notas de classes altas até piores que as de classes mais baixas.

Com testes de diferença de média, observamos que não só o tipo de escola, mas que também as outras variáveis são significativas na nota final do participante do Enem, diretamente ou mediada pelo Critério Brasil. Mesmo usando dados apenas do ano 2021, os DAGs propostos pelos algoritmos estão indicando as mesmas causalidades que foram encontradas em trabalhos anteriores, que são a renda e escolaridade dos pais (LOBO *et al*., 2017; GREMAUD *et al*., 2007; Barros *et al*., 2001), essas sendo explicadas pelas outras variáveis do modelo, cor/raça e região geográfica do participante.

Na função `hc`, a relação “classe de renda”–“tipo de escola” aconteceu naturalmente, diferente dos algoritmos baseados em restrição, e a direção também foi de renda explicando o tipo de escola. Os algoritmos do pacote `pcalg` não foram capazes de identificar a orientação dessa relação quando forçada a relação entre elas. Mesmo que essa relação “classe de renda”–“tipo de escola” só apareça quando forçada, cerca de 93% dos estudantes da classe A e cerca de 65% dos estudantes da classe B do Critério Brasil responderam que estudam em escolas particulares. Com a Lei de Cotas, vem ocorrendo uma migração de alunos da rede particular para a rede pública, tanto por parte de alunos com baixa quanto por alta classe de renda familiar (MELLO, 2021; SENKEVICS & MELLO, 2019), o que pode explicar a ausência de relação entre a classe do Critério Brasil e o tipo de escola nos modelos baseados em restrição. Como também evidenciado por Borges (2021), o tipo de escola não é suficiente para explicar o acesso ao ensino superior, resultante de uma boa nota no Enem. Desta forma, é necessário que o aluno busque complementar sua formação,

como cursos preparatórios ou de língua estrangeira, sendo um fator limitador para aqueles de renda ou classes mais baixas, justificando o tipo de escola não ser mediador para a classe de renda na nota do participante do Enem 2021, mas sim a nota sendo efeito comum do tipo de escola e da classe de renda.

Os modelos propostos sem e com as mesmas relações forçadas, gerados pelos algoritmos PC e FCI, foram equivalentes por Markov. Podemos observar que para as mesmas restrições impostas, onde houve incerteza na direção da relação na função `pc`, essa direção foi representada com uma possível causa comum não observada na `fci`.

Para os três algoritmos, os modelos sempre propuseram que a nota era explicada pelas variáveis que estavam relacionadas a ela, nunca o contrário. As variáveis que explicam a nota, em todos os modelos aqui gerados, foram o tipo de escola (ESC) e classe do Critério Brasil (CRIT). Pela função `hc`, não só a escolaridade da mãe (Q002) influencia diretamente na nota, sem necessidade de forçar relação, mas também essas três variáveis foram confundidoras. Para `pc` e `fci`, a escolaridade da mãe explica a nota através da classe de poder aquisitivo (CRIT), passando a explicar diretamente a nota após as remoções das variáveis sexo (SEX), localidade da escola (LOC) e acesso à internet em casa (NET).

A variável indicadora CRIT, mesmo quando forçando a relação com o tipo de escola, é explicada pelas outras variáveis, tanto nos algoritmos baseados em restrição quanto nos baseados em escore. Nos algoritmos baseados em restrição, as variáveis sexo do participante (SEX) e localidade da escola (LOC) não tiveram relação com nenhuma das outras variáveis selecionadas. A variável acesso à internet em casa (NET) também foi removida pois, em quase todos os modelos gerados, não explicava a nota final, seja direta ou indiretamente. Nos algoritmos baseados em escore, porém, algumas variáveis estavam sendo explicadas por outras de forma não esperada, como a classe do Critério Brasil explicando o sexo do participante ou o tipo de escola explicando a cor/raça do participante. A remoção dessas ligações e os modelos subsequentes estão de acordo com os trabalhos anteriores (LOBO *et al*., 2017; GREMAUD *et al*., 2007; BARROS *et al*., 2001).

Estudos anteriores (*e.g.*, LOBO *et al*., 2017; GREMAUD *et al*., 2007; MELO *et al*., 2018; MENEZES-FILHO, 2007) também apresentaram os níveis de escolaridade dos pais do estudante como influentes no desempenho escolar, aqui atuando

indiretamente através da classe do Critério Brasil. Quando mantivemos a escolaridade do pai e a escolaridade da mãe, as duas sempre se relacionam, mas nunca as duas escolaridades explicando a nota do participante simultânea e diretamente, exceto quando forçada as relações. Ao remover a escolaridade do pai para geração de modelos, os mesmos são iguais ao remover a escolaridade da mãe e vice-versa.

Fatores como renda influenciam no desempenho escolar do aluno, impactando na nota do Enem e limitando o acesso ao ensino superior. A formação dos pais pode até ser mais influente que a própria renda familiar no desempenho escolar do estudante segundo alguns autores (*e.g.* BARROS *et al*., 2001); uma reação em cadeia causada pelas mesmas variáveis. Essas mesmas variáveis também estão relacionadas à evasão do ensino superior (SACCARO *et al*., 2019; CHEN & SOLDNER, 2013), não se limitando apenas nos desempenhos no ensino médio e no Enem. Então, mesmo que existam incentivos e oportunidades para ingresso no ensino superior, como o Prouni (BRASIL, 2004) e a Lei de Cotas, nº 12.711/2012 (BRASIL, 2012), esse aluno pode não conseguir chegar até o final da graduação, podendo causar uma reação em cadeia.

5 CONCLUSÃO

A partir de nossos resultados, temos evidências de que o tipo de escola, a classe de renda familiar e a escolaridade dos pais são fatores mais significativos no desempenho do participante do Enem. Mesmo utilizando apenas os dados do ano de 2021, esses resultados estão de acordo com a literatura consultada. Nos modelos gerados, o tipo de escola nem sempre foi diretamente relacionado à classe do Critério Brasil, mas as demais variáveis finais sim. O trabalho aqui apresentado é apenas para propor modelos causais para fatores que podem influenciar na nota do Enem, visando auxiliar trabalhos futuros que busquem definir os níveis de influência dessas variáveis. A descoberta causal é muito útil para o entendimento da interação entre as variáveis como um todo, mas ainda sendo necessário saber os motivos de tais interações e quais são os possíveis fatores não observados que afetam os resultados que temos acesso, também sendo oportunidade para estudo de outras áreas além da parte estatística.

Este trabalho uniu, de maneira formal e também através de uma linguagem apropriada, duas fontes de informações fundamentais para a Inferência Causal: conhecimento prévio e evidências empíricas. Também tornou explícitas, através da linguagem apropriada, as suposições sobre as relações causais estudadas. Trabalhos anteriores se baseiam apenas em percepções e correlações empíricas, sem um modelo que estruture estas possíveis relações. Os DAGs são uma maneira de modelar as suposições sobre a estrutura causal. Por consequência, ajuda a orientar o ajuste de modelos que pretendam estimar efeitos causais do tipo de escola na nota do Enem, explicitando as escolhas que levaram à crença de que as possíveis fontes de viés foram controladas. Com isso, a partir deste trabalho, podemos propor, por exemplo que ao se considerar um modelo que pretenda estimar o efeito do tipo de escola a nota do Enem, o sexo, a localidade e o acesso à internet não devem entrar como variáveis de controle. Além do mais, encontramos que a classe do Critério Brasil pode ser considerada mediadora de cor e região, isto é, que estas variáveis não deveriam estar presentes neste modelo uma vez que essa classe, ou outra variável que a represente, também esteja. E, por fim, que o tipo de escola, a classe de renda (ou equivalente) e a escolaridade do pai ou da mãe devem ser fatores causais diretamente explicativos da nota no Enem.

REFERÊNCIAS

ANDREWS, R. M.; FORAITA, R.; DIDELEZ, V.; WITTE, J. **A practical guide to causal discovery with cohort data.** 2021. DOI 10.48550/ARXIV.2108.13395. Disponível em: https://arxiv.org/abs/2108.13395.

ANKAN, A.; WORTEL, I. M. N.; TEXTOR, J. Testing Graphical Causal Models Using the R Package “dagitty”. **Current Protocols**, v. 1, n. 2, p. e45, 2021. https://doi.org/10.1002/cpz1.45.

BARROS, R. P. D.; MENDONÇA, R.; SANTOS, D. D. D.; QUINTAES, G. **Determinantes do desempenho educacional no Brasil.** v. 31, n. 1, 2001.

BERETTA, S.; CASTELLI, M.; GONÇALVES, I.; HENRIQUES, R.; RAMAZZOTTI, D. Learning the Structure of Bayesian Networks: A Quantitative Assessment of the Effect of Different Algorithmic Schemes. **Complexity**, v. 2018, p. 1–12, 12 set. 2018. https://doi.org/10.1155/2018/1591878.

BORGES, J. L. das G.; CARNIELLI, B. L. Educação e estratificação social no acesso à universidade pública. **Cadernos de Pesquisa**, v. 35, n. 124, p. 113–139, abr. 2005. https://doi.org/10.1590/S0100-15742005000100007.

BRASIL. Lei nº 11.096, de 13 de janeiro de 2005. Institui o Programa Universidade para Todos (PROUNI), regula a atuação de entidades beneficentes de assistência social no ensino superior, altera a Lei n. 10.981, de 9 de julho de 2004, e dá outras providências. **Diário Oficial da União**, Brasília, 14 jan. 2005.

BRASIL. Lei nº 12.711, de 29 de agosto de 2012. Dispõe sobre o ingresso nas universidades federais e nas instituições federais de ensino técnico de nível médio e dá outras providências. **Diário Oficial da União**, Brasília, 30 ago. 2012.

CENTOLA, D.; MACY, M. Complex Contagions and the Weakness of Long Ties. **American Journal of Sociology**, v. 113, n. 3, p. 702–734, nov. 2007. https://doi.org/10.1086/521848.

CINELLI, C.; FORNEY, A.; PEARL, J. A Crash Course in Good and Bad Controls. **Sociological Methods & Research**, p. 00491241221099552, 20 maio 2022. https://doi.org/10.1177/00491241221099552.

COLOMBO, D.; MAATHUIS, M. H.; KALISCH, M.; RICHARDSON, T. S. Learning high-dimensional directed acyclic graphs with latent and selection variables. **The Annals of Statistics**, v. 40, n. 1, 1 fev. 2012. DOI 10.1214/11-AOS940. Disponível em: http://arxiv.org/abs/1104.5617.

CUNNINGHAM, S. **Causal Inference: The Mixtape**. Yale University Press, 2021. DOI 10.2307/j.ctv1c29t27. Disponível em: https://www.jstor.org/stable/j.ctv1c29t27.

DUNCAN, O. D. **Introduction to Structural Equation Models**. Elsevier, 2014.

FORAITA, R.; WITTE, J. **micd: Multiple Imputation in Causal Graph Discovery**. 5 set. 2022. Disponível em: https://CRAN.R-project.org/package=micd.

GLYMOUR, C.; ZHANG, K.; SPIRTES, P. Review of Causal Discovery Methods Based on Graphical Models. **Frontiers in Genetics**, v. 10, p. 524, 4 jun. 2019. https://doi.org/10.3389/fgene.2019.00524.

GREMAUD, A. P.; FELICIO, F. de; BIONDI, R. L. Indicador de efeito escola: **Textos para discussão**, n. 27, p. 33–33, 2007.

HANUSHEK, E. A. The Impact of Differential Expenditures on School Performance. **Educational Researcher**, v. 18, n. 4, p. 45–62, maio 1989. https://doi.org/10.3102/0013189X018004045.

HILL CLIMBING ALGORITHM IN AI - JAVATPOINT. [*s. d.*]. **javaTpoint**. Disponível em: https://www.javatpoint.com/hill-climbing-algorithm-in-ai.

JACOB, M. E.; GANGULI, M. Epidemiology for the clinical neurologist. **Handbook of Clinical Neurology**, v. 138, p. 3–16, 2016. https://doi.org/10.1016/B978-0-12-802973-2.00001-X.

KALISCH, Markus. Estimating High-Dimensional Directed Acyclic Graphs with the PC-Algorithm. **Journal of Machine Learning Research**, v. 8, p. 613–636, 2007.

KALISCH, Markus; HAUSER, A.; MAECHLER, M.; COLOMBO, D.; ENTNER, D.; HOYER, P.; HYTTINEN, A.; PETERS, J.; ANDRI, N.; PERKOVIC, E.; NANDY, P.; RUETIMANN, P.; STEKHOVEN, D.; SCHUERCH, M.; EIGENMANN, M.; HENCKEL, L.; MOOIJ, J. **pcalg: Methods for Graphical Models and Causal Inference**. 4 out. 2022. Disponível em: https://CRAN.R-project.org/package=pcalg.

KALISCH, Markus; MÄCHLER, M.; COLOMBO, D.; MAATHUIS, M. H.; BÜHLMANN, P. Causal Inference Using Graphical Models with the R Package pcalg. **Journal of Statistical Software**, v. 47, p. 1–26, 17 maio 2012. https://doi.org/10.18637/jss.v047.i11.

LEARNING, G. An Introduction to Hill Climbing Algorithm in AI (Artificial Intelligence). **Great Learning Blog: Free Resources what Matters to shape your Career!** 22 maio 2020. Disponível em: https://www.mygreatlearning.com/blog/an-introduction-to-hill-climbing-algorithm/.

LOBO, G. D.; CASSUCE, F. C. C.; CIRINO, J. F. Avaliação do desempenho escolar dos estudantes da região nordeste que realizaram o Enem: Uma análise com **Modelos Hierárquicos**. v. 38, 1 jan. 2017.

MELLO, U. Affirmative Action and the Choice of Schools. **BSE Working Papers**, 2021.

MENEZES-FILHO, N. A. **Os determinantes do desempenho escolar do Brasil.** 2007. Disponível em: https://repositorio.usp.br/item/001624821.

OLIVEIRA, A. F. G. TESTES ESTATÍSTICOS PARA COMPARAÇÃO DE MÉDIAS. **Revista Eletrônica Nutritime**, v. 5, n. 6, p. 777–788, 2008.

PEARL, J. 3. The Foundations of Causal Inference. **Sociological Methodology**, v. 40, n. 1, p. 75–149, ago. 2010. https://doi.org/10.1111/j.1467-9531.2010.01228.x.

PEARL, J.; GLYMOUR, M.; JEWELL, N. P. **Causal Inference in Statistics: A Primer**. John Wiley & Sons, 2016.

PEREIRA, R. H. M.; GONCALVES, C. N.; ARAUJO, P. H. F. de; CARVALHO, G. D.; ARRUDA, R. A. de; NASCIMENTO, I.; COSTA, B. S. P. da; CAVEDO, W. S.; ANDRADE, P. R.; SILVA, A. da; BRAGA, C. K. V.; SCHMERTMANN, C.; SAMUEL-ROSA, A.; FERREIRA, D.; SARAIVA, M.; RESEARCH, I.-I. for A. E. **geobr: Download**

Official Spatial Data Sets of Brazil. 16 ago. 2022. Disponível em: https://CRAN.R-project.org/package=geobr.

PERKOVIĆ, E.; TEXTOR, J.; KALISCH, M. Complete Graphical Characterization and Construction of Adjustment Sets in Markov Equivalence Classes of Ancestral Graphs. **Journal of Machine Learning Research**, v. 18, p. 1–62, 18 maio 2018.

ROHEKAR, R. Y.; NISIMOV, S.; GURWICZ, Y.; NOVIK, G. Iterative Causal Discovery in the Possible Presence of Latent Confounders and Selection Bias. **Conference on Neural Information Processing Systems**, n. 35, 2021.

ROJAS-CARULLA, M.; BARONI, M.; LOPEZ-PAZ, D. Causal Discovery Using Proxy Variables. **arXiv preprint arXiv:1702.07306**, 23 fev. 2017. Disponível em: http://arxiv.org/abs/1702.07306.

RUBIN, D. B. [On the Application of Probability Theory to Agricultural Experiments. Essay on Principles. Section 9.] Comment: Neyman (1923) and Causal Inference in Experiments and Observational Studies. **Statistical Science**, v. 5, n. 4, p. 472–480, 1990.

SCHEINES, R. An Introduction to Causal Inference. *In*: MCKIM, V.; TURNER, S. (org.) **Causality in Crisis?** 2. ed. University of Notre Dame Press, 1997. v. 30, p. 185–199.

SCUTARI, M.; SILANDER, T.; NESS, R. **bnlearn: Bayesian Network Structure Learning, Parameter Learning and Inference**. 21 set. 2022. Disponível em: https://CRAN.R-project.org/package=bnlearn.

SENKEVICS, A. S.; MELLO, U. M. O PERFIL DISCENTE DAS UNIVERSIDADES FEDERAIS MUDOU PÓS-LEI DE COTAS? **Cadernos de Pesquisa**, v. 49, p. 184–208, 10 jul. 2019. https://doi.org/10.1590/198053145980.

SHEN, X.; MA, S.; VEMURI, P.; SIMON, G. Challenges and Opportunities with Causal Discovery Algorithms: Application to Alzheimer's Pathophysiology. **Scientific Reports**, v. 10, n. 1, p. 2975, 19 fev. 2020. https://doi.org/10.1038/s41598-020-59669-x.

SILVA, A. A. M. da. **Introdução à inferência causal em epidemiologia: uma abordagem gráfica e contrafatual**. Rio de Janeiro: Editora Fiocruz, 2021. Disponível em: https://pesquisa.bvsalud.org/portal/resource/pt/biblio-1369694.

SLOWIKOWSKI, K.; SCHEP, A.; HUGHES, S.; DANG, T. K.; LUKAUSKAS, S.; IRISSON, J.-O.; KAMVAR, Z. N.; RYAN, T.; CHRISTOPHE, D.; HIROAKI, Y.; GRAMME, P.; ABDOL, A. M.; BARRETT, M.; CANNOODT, R.; KRASSOWSKI, M.; CHIRICO, M.; APHALO, P. **ggrepel: Automatically Position Non-Overlapping Text Labels with "ggplot2"**. 6 nov. 2022a. Disponível em: https://CRAN.R-project.org/package=ggrepel.

SPIRTES, P. Introduction to Causal Inference. **Journal of Machine Learning Research**, v. 11, p. 1643–1662, 2010.

SPIRTES, P.; GLYMOUR, C. An Algorithm for Fast Recovery of Sparse Causal Graphs. **Social Science Computer Review**, v. 9, n. 1, p. 62–72, abr. 1991. https://doi.org/10.1177/089443939100900106.

SPIRTES, P.; GLYMOUR, C. N.; SCHEINES, R.; HECKERMAN, D. **Causation, Prediction, and Search**. MIT Press, 2000.

TALEBI, S. Causal Discovery. **Medium**. 22 dez. 2022. Disponível em: https://towardsdatascience.com/causal-discovery-6858f9af6dcb.

TUKEY, J. W. Comparing Individual Means in the Analysis of Variance. **Biometrics**, v. 5, n. 2, p. 99–114, 1949. https://doi.org/10.2307/3001913.

WANG, Y. Analysis of the Max-Min Hill-Climbing Algorithm. *In*: 2018 INTERNATIONAL CONFERENCE ON TRANSPORTATION & LOGISTICS, INFORMATION & COMMUNICATION, SMART CITY (TLICSC 2018), dez. 2018. Atlantis Press, dez. 2018. p. 509–511. DOI 10.2991/tlicsc-18.2018.82. Disponível em: https://www.atlantis-press.com/proceedings/tlicsc-18/25907289.

WICKHAM, H.; CHANG, W.; HENRY, L.; PEDERSEN, T. L.; TAKAHASHI, K.; WILKE, C.; WOO, K.; YUTANI, H.; DUNNINGTON, D.; RSTUDIO. **ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics**. 4 nov. 2022. Disponível em: https://CRAN.R-project.org/package=ggplot2.

WICKHAM, H.; FRANÇOIS, R.; HENRY, L.; MÜLLER, K.; RSTUDIO. **dplyr: A Grammar of Data Manipulation**. 1 set. 2022. Disponível em: https://CRAN.R-project.org/package=dplyr.

WRIGHT, S. Correlation and Causation. **Journal of Agricultural Research**, v. 20, n. 7, p. 557-585., 1921.

ZHANG, J. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. **Artificial Intelligence**, v. 172, n. 16, p. 1873–1896, 1 nov. 2008. https://doi.org/10.1016/j.artint.2008.08.001.

APÊNDICE A – Frequências e porcentagens nas variáveis selecionadas

Tabela 2 – Critério Brasil

Critério Brasil	Freq.	%
A	33.321	6,56%
B1	27.721	5,46%
B2	73.331	14,44%
C1	79.742	15,70%
C2	101.102	19,91%
DE	192.594	37,93%
Total	**507.811**	**100%**

FFon

Tabela 3 – Escolaridade do pai (Q001)

Q001	Freq.	%
A - Nunca estudou	10.962	2,16%
B - Não completou a 4ª série/5º ano do Ensino Fundamental	58.522	11,52%
C - Completou a 4ª série/5º ano, mas não completou a 8ª série/9º ano do Ensino Fundamental	59.427	11,70%
D - Completou a 8ª série/9º ano do Ensino Fundamental, mas não completou o Ensino Médio	58.760	11,70%
E - Completou o Ensino Médio, mas não completou a Faculdade	161.026	31,71%
F - Completou a Faculdade, mas não completou a Pós-graduação	65.299	12,86%
G - Completou a Pós-graduação	52.017	10,24%
H - Não sei	41.798	8,23%
Total	**507.811**	**100%**

Tabela 4 – Escolaridade da mãe (Q002)

Q002	Freq.	%
A - Nunca estudou	5.708	1,12%
B - Não completou a 4ª série/5º ano do Ensino Fundamental	36.915	7,27%
C - Completou a 4ª série/5º ano, mas não completou a 8ª série/9º ano do Ensino Fundamental	46.245	9,11%
D - Completou a 8ª série/9º ano do Ensino Fundamental, mas não completou o Ensino Médio	57.433	11,31%
E - Completou o Ensino Médio, mas não completou a Faculdade	183.685	36,17%
F - Completou a Faculdade, mas não completou a Pós-graduação	82.895	16,32%
G - Completou a Pós-graduação	80.301	15,81%
H - Não sei	14.629	2,88%
Total	**507.811**	**100%**

Tabela 5 – Localização da escola

Localização da escola	Freq.	%
Urbana	495.222	97,52%
Rural	12.589	2,47%
Total	**507.811**	**100%**

Tabela 6 – Cor/raça do participante

Cor/raça	Freq.	%
Branca	245.607	48,37%
Parda	195.768	38,55%
Preta	44.499	8,76%
Amarela	9.600	1,89%
Indígena	2.096	0,41%
Não declarado	10.241	2,02%
Total	**507.811**	**100%**

Tabela 7– Sexo do participante

Localização da escola	Freq.	%
Feminino	288.390	56,8%
Masculino	219.390	43,2%
Total	**507.811**	**100%**

Tabela 8 – Acesso à internet em casa (Q025)

Q025	Freq.	%
Sim	469.427	92,44%
Não	38.384	7,56%
Total	**507.811**	**100%**

Tabela 9 – Região da escola do participante

Região	Freq.	%
Sudeste	185.670	36,56%
Nordeste	165.958	32,68%
Sul	65.790	12,96%
Centro Oeste	47.124	9,28%
Norte	43.269	8,52%
Total	**507.811**	**100%**

APÊNDICE B – Script R

```
```{r}
####RECOMENDO COPIAR E COLAR EM UM RMARKDOWN
####também disponível em https://github.com/brunohr/TCC
```

```{r}
"%!in%" = Negate("%in%")

library(tidyverse)
# library(readr)
# library(dplyr)
# library(ggplot2)
library(ggrepel)
library(ggpp)
library(reshape2)
library(geobr)
library(descr)
library(lavaan)
library(pcalg)
# devtools::install_github("bips-hb/micd")
library(micd)
# BiocManager::install(c('graph', 'RBGL', 'Rgraphviz'))
library(bnlearn)
```

DADOS

LEITURA DOS DADOS

```{r}
# # aumentando tempo de conexão do R com o servidor (segundos)
options(timeout = max(600, getOption("timeout")))
#
# #baixando o arquivo zipado direto do INEP
temp <- tempfile()
download.file("https://download.inep.gov.br/microdados/microdados_en
em_2021.zip", temp)
# # descompactando o arquivo
data <- read.table(unz(temp, "DADOS/MICRODADOS_ENEM_2021.csv"))

enem0 =
  read_delim(unz("microdados_enem_2021.zip",
"DADOS/MICRODADOS_ENEM_2021.csv"),
             delim = ";",
             escape_double = FALSE,
             col_types = cols(
               NU_INSCRICAO = col_factor(),
               TP_FAIXA_ETARIA = col_factor(
                 c("1", "2", "3", "4", "5", "6", "7", "8", "9",
"10", "11", "12", "13", "14", "15", "16", "17", "18", "19", "20")),
               TP_SEXO = col_factor(c("M", "F")),
```
```

```
               TP_ESTADO_CIVIL = col_factor(c("0", "1", "2", "3",
"4", "5")),
               TP_NACIONALIDADE = col_factor(c("0", "1", "2", "3",
"4")),
               TP_ST_CONCLUSAO = col_factor(c("1", "2", "3", "4")),
               TP_ANO_CONCLUIU = col_factor(c("0", "1", "2", "3",
"4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15")),
               TP_ESCOLA = col_factor(c("1", "2", "3")),
               TP_ENSINO = col_factor(c("1", "2")),
               IN_TREINEIRO = col_factor(c("1", "0")),
               TP_DEPENDENCIA_ADM_ESC = col_factor(c("1", "2", "3",
"4")),
               TP_LOCALIZACAO_ESC = col_factor(c("1", "2")),
               TP_SIT_FUNC_ESC = col_factor(c("1", "2", "3", "4")),
               CO_MUNICIPIO_PROVA = col_factor(),
               CO_UF_PROVA = col_factor(),
               TP_PRESENCA_CN = col_factor(c("0", "1", "2")),
               TP_PRESENCA_CH = col_factor(c("0", "1", "2")),
               TP_PRESENCA_LC = col_factor(c("0", "1", "2")),
               TP_PRESENCA_MT = col_factor(c("0", "1", "2")),
               CO_PROVA_CN = col_factor(c("909", "910", "911",
"912",
                                          "915", "916", "917",
"989",
                                          "990", "991", "992",
"996",
                                          "1011", "1012", "1013",
"1014",
                                          "1045", "1046", "1047",
"1048", "1052")),
               CO_PROVA_CH = col_factor(c("879", "880", "881",
"882",
                                          "885", "886", "887",
"959",
                                          "960", "961", "962",
"966",
                                          "999", "1000", "1001",
"1002",
                                          "1015", "1016", "1017",
"1018", "1022")),
               CO_PROVA_LC = col_factor(c("889", "890", "891",
"892",
                                          "895", "896", "897",
"969",
                                          "970", "971", "972",
"976",
                                          "1003", "1004", "1005",
"1006",
                                          "1025", "1026", "1027",
"1028", "1032")),
               CO_PROVA_MT = col_factor(c("899", "900", "901",
"902",
                                          "905", "906", "907",
"979",
                                          "980", "981", "982",
"986",
```

```
                                          "1007", "1008", "1009",
"1010",
                                          "1035", "1036", "1037",
"1038", "1042")),
               TP_LINGUA = col_factor(c("0", "1")),
               TP_STATUS_REDACAO = col_factor(c("1", "2", "3", "4",
"5",
                                                "6", "7", "8",
"9")),
               Q001 = col_factor(c("A", "B", "C", "D", "E", "F",
"G", "H")),
               Q002 = col_factor(c("A", "B", "C", "D", "E", "F",
"G", "H")),
               Q003 = col_factor(c("A", "B", "C", "D", "E", "F"),
                                 ordered = T),
               Q004 = col_factor(c("A", "B", "C", "D", "E", "F"),
                                 ordered = T),
               Q005 = col_factor(c("0", "1", "2", "3", "4", "5",
"6", "7",
                                   "8", "9", "10", "11", "12", "13",
"14",
                                   "15", "16", "17", "18", "19",
"20"),
                                 ordered = T),
               Q006 = col_factor(c("A", "B", "C", "D", "E", "F",
"G", "H",
                                   "I", "J", "K", "L", "M", "N",
"O", "P", "Q"),
                                 ordered = T),
               Q007 = col_factor(c("A", "B", "C", "D"),
                                 ordered = T),
               Q008 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q009 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q010 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q011 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q012 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q013 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q014 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q015 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q016 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q017 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q018 = col_factor(c("A", "B"),
                                 ordered = T),
               Q019 = col_factor(c("A", "B", "C", "D", "E"),
                                 ordered = T),
               Q020 = col_factor(c("A", "B"),
```

```
                                ordered = T),
              Q021 = col_factor(c("A", "B"),
                                ordered = T),
              Q022 = col_factor(c("A", "B", "C", "D", "E"),
                                ordered = T),
              Q023 = col_factor(c("A", "B"),
                                ordered = T),
              Q024 = col_factor(c("A", "B", "C", "D", "E"),
                                ordered = T),
              Q025 = col_factor(c("A", "B"),
                                ordered = T)),
            trim_ws = TRUE)
unlink(temp)

enem1 = enem0 %>%
  mutate(NOTA_FINAL = (NU_NOTA_REDACAO + NU_NOTA_CH + NU_NOTA_CN +
## nota final
                         NU_NOTA_LC +  NU_NOTA_MT)/5, ## nota final

         criter_aux = ifelse(Q008 == "A", 0, ## banheiro
                             ifelse(Q008 == "B", 3, ## banheiro
                                    ifelse(Q008 == "C", 7, ##
banheiro
                                           ifelse(Q008 == "D", 10,
## banheiro
                                                  ifelse(Q008 ==
"E", 14, 0))))) +
           ifelse(Q007 == "A", 0, ## empregado doméstico
                  ifelse(Q007 == "B", 3, ## empregado doméstico
                         ifelse(Q007 == "C", 7, ## empregado
doméstico
                                ifelse(Q007 == "D", 10, ## empregado
doméstico
                                       ifelse(Q007 == "E", 13,
0))))) +
           ifelse(Q010 == "A", 0, ## carro
                  ifelse(Q010 == "B", 3, ## carro
                         ifelse(Q010 == "C", 5, ## carro
                                ifelse(Q010 == "D", 8, ## carro
                                       ifelse(Q010 == "E", 11,
0))))) + ## carro
           ifelse(Q024 == "A", 0, ## carro
                  ifelse(Q024 == "B", 3, ## carro
                         ifelse(Q024 == "C", 6, ## carro
                                ifelse(Q024 == "D", 8, ## carro
                                       ifelse(Q024 == "E", 11,
0))))) + ## computador
           ifelse(Q017 == "A", 0, ## lava louça
                  ifelse(Q017 == "B", 3, ## lava louça
                         ifelse(Q017 == "C", 6, ## lava louça
                                ifelse(Q017 == "D", 8, ## lava louça
                                       ifelse(Q008 == "E", 11,
0))))) + ## lava louça
           ifelse(Q012 == "A", 0, ## geladeira
                  ifelse(Q012 == "B", 2, ## geladeira
                         ifelse(Q012 == "C", 3, ## geladeira
```

```
                                    ifelse(Q012 == "D", 5, ## geladeira
                                           ifelse(Q012 == "E", 5, 0)))))
+ ## geladeira
           ifelse(Q013 == "A", 0, ## freezer
                  ifelse(Q013 == "B", 2, ## freezer
                         ifelse(Q013 == "C", 4, ## freezer
                                ifelse(Q013 == "D", 6, ## freezer
                                       ifelse(Q013 == "E", 6, 0)))))
+ ## freezer
           ifelse(Q014 == "A", 0, ## lava roupa
                  ifelse(Q014 == "B", 2, ## lava roupa
                         ifelse(Q014 == "C", 4, ## lava roupa
                                ifelse(Q014 == "D", 6, ## lava roupa
                                       ifelse(Q014 == "E", 6, 0)))))
+ ## lava roupa
           ifelse(Q013 == "A", 0, ## dvd
                  ifelse(Q013 == "B", 1, ## dvd
                         ifelse(Q013 == "C", 3, ## dvd
                                ifelse(Q013 == "D", 4, ## dvd
                                       ifelse(Q013 == "E", 6, 0)))))
+ ## dvd
           ifelse(Q016 == "A", 0, ## microondas
                  ifelse(Q016 == "B", 2, ## microondas
                         ifelse(Q016 == "C", 4, ## microondas
                                ifelse(Q016 == "D", 4, ## microondas
                                       ifelse(Q016 == "E", 4, 0)))))
+ ## microondas
           ifelse(Q011 == "A", 0, ## motocicleta
                  ifelse(Q011 == "B", 1, ## motocicleta
                         ifelse(Q011 == "C", 3, ## motocicleta
                                ifelse(Q011 == "D", 3, ##
motocicleta
                                       ifelse(Q011 == "E", 3, 0)))))
+ ## motocicleta
           ifelse(Q015 == "A", 0, ## secadora roupa
                  ifelse(Q015 == "B", 2, ## secadora roupa
                         ifelse(Q015 == "C", 2, ## secadora roupa
                                ifelse(Q015 == "D", 2, ## secadora
roupa
                                       ifelse(Q015 == "E", 2, 0)))))
+ ## secadora roupa
           ifelse(Q001 %in% c("A", "B"), 0, ## escolarida chefe da
casa
                  ifelse(Q001 %in% c("C"), 1, ## escolarida chefe da
casa
                         ifelse(Q001 %in% c("D"), 2, ## escolarida
chefe da casa
                                ifelse(Q001 %in% c("E"), 4, ##
escolarida chefe da casa
                                       ifelse(Q001 %in% c("F", "G"),
7, 0)))))
  ) %>%
  mutate(CRITERIO = factor(ifelse(criter_aux >= 45, "A",
                                  ifelse(criter_aux >= 38, "B1",
                                         ifelse(criter_aux >= 29,
"B2",
```

```
                                                    ifelse(criter_aux >=
23, "C1",

ifelse(criter_aux >= 17, "C2",

ifelse(criter_aux >= 0,

"DE"))))))),
                              ordered = T),
         CRITERIO2 = factor(ifelse(criter_aux >= 45, "A",
                                   ifelse(criter_aux >= 29, "B",
                                          ifelse(criter_aux >= 17,
"C",
                                                 ifelse(criter_aux
>= 0, "DE")))),
                               ordered = T),
         REGIAO = factor(
           ifelse(SG_UF_ESC %in% c("RO", "AC", "AM", "RR", "PA",
"AP", "TO"), "Norte",
                  ifelse(SG_UF_ESC %in% c("MA", "PI", "CE", "RN",
"PB", "PE", "AL", "SE", "BA"),
                         "Nordeste",
                         ifelse(SG_UF_ESC %in% c("MG", "ES", "RJ",
"SP"), "Sudeste",
                                ifelse(SG_UF_ESC %in% c("PR", "SC",
"RS"), "Sul",
                                       ifelse(SG_UF_ESC %in% c("MS",
"MT", "GO", "DF"), "Centro-Oeste",
                                              NA))))))
  )

enem1$IN_TREINEIRO <- factor(enem1$IN_TREINEIRO, levels = c(1,0),
labels=c('Sim', 'Não'))

enem1$TP_DEPENDENCIA_ADM_ESC <- factor(enem1$TP_DEPENDENCIA_ADM_ESC,
levels = c(1,2,3,4),
                                       labels=c('Federal',
                                                'Estadual',
                                                'Municipal',
                                                'Privada'))

enem1$TP_LOCALIZACAO_ESC <- factor(enem1$TP_LOCALIZACAO_ESC, levels
= c(1,2), labels=c('Urbana', 'Rural'))

enem1$TP_SIT_FUNC_ESC <- factor(enem1$TP_SIT_FUNC_ESC, levels =
c(1,2,3,4),
                                labels=c('Em atividade',
                                         'Paralisada',
                                         'Extinta',
                                         'Escola extinta em anos
anteriores'))

enem1$TP_SEXO <- factor(enem1$TP_SEXO, levels = c('M', 'F'),
labels=c('Masculino', 'Feminino'))
```

```
enem1$TP_COR_RACA <- factor(enem1$TP_COR_RACA, levels =
c(0,1,2,3,4,5,6),
                            labels=c('Não declarado',
                                     'Branca', 'Preta',
                                     'Parda', 'Amarela',
                                     'Indígena',
                                     'Não dispõe da informação'))

enem1$TP_ST_CONCLUSAO <- factor(enem1$TP_ST_CONCLUSAO, levels =
c(1,2,3,4),
                                labels=c('Já concluí o Ensino
Médio',
                                         'Estou cursando e
concluirei o Ensino Médio em 2021',
                                         'Estou cursando e
concluirei o Ensino Médio após 2021',
                                         'Não concluí e não estou
cursando o Ensino Médio'))

enem1$TP_ANO_CONCLUIU <- factor(enem1$TP_ANO_CONCLUIU, levels =
c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15),
                                labels=c('Não informado', '2020',
'2019', '2018', '2017', '2016',
                                         '2015', '2014', '2013',
                                         '2012', '2011', '2010',
                                         '2009', '2008', '2007',
                                         'Anterior a 2007'))

enem1$TP_ESCOLA <- factor(enem1$TP_ESCOLA, levels = c(1,2,3),
                          labels=c('Não respondeu',
                                   'Pública',
                                   'Privada'))

enem1$TP_ENSINO <- factor(enem1$TP_ENSINO, levels = c(1,2),
                          labels=c('Ensino Regular',
                                   'Educação Especial - Modalidade
Substitutiva'))


enem1$TP_PRESENCA_CN <- factor(enem1$TP_PRESENCA_CN, levels =
c(0,1,2),
                               labels=c('Faltou à prova',
                                        'Presente na prova',
                                        'Eliminado na prova'))

enem1$TP_PRESENCA_CH <- factor(enem1$TP_PRESENCA_CH, levels =
c(0,1,2),
                               labels=c('Faltou à prova',
                                        'Presente na prova',
                                        'Eliminado na prova'))

enem1$TP_PRESENCA_LC <- factor(enem1$TP_PRESENCA_LC, levels =
c(0,1,2),
                               labels=c('Faltou à prova',
                                        'Presente na prova',
                                        'Eliminado na prova'))
```

```
enem1$TP_PRESENCA_MT <- factor(enem1$TP_PRESENCA_MT, levels =
c(0,1,2),
                               labels=c('Faltou à prova',
                                        'Presente na prova',
                                        'Eliminado na prova'))

enem1$TP_STATUS_REDACAO <- factor(enem1$TP_STATUS_REDACAO, levels =
c(1,2,3,4,5,6,7,8,9),
                                  labels=c('Sem problemas',
                                           'Anulada', 'Cópia Texto
Motivador',
                                           'Em Branco', 'Fere
Direitos Humanos',
                                           'Fuga ao tema',
                                           'Não atendimento ao
tipo',
                                           'Texto insuficiente',
                                           'Parte desconectada'))

# Na sua residência tem acesso à Internet?
enem1$Q025 <- factor(enem1$Q025, levels = c('A', 'B'),
                     labels=c('Não', 'Sim'))
```


FILTROS

```{r}
enem = enem1 %>%
  filter(
    #considerando presença na prova de linguagens
    TP_PRESENCA_LC == 1, #"Presente na prova"

    #considerando presença na prova de matemática
    TP_PRESENCA_MT == 1, #"Presente na prova"

    #considerando presença na prova de ciências da natureza
    TP_PRESENCA_CN == 1, #"Presente na prova"

    #considerando presença na prova de ciências humanas
    TP_PRESENCA_CH == 1, #"Presente na prova"

    #considerando apenas escolas em atividade
    TP_SIT_FUNC_ESC == 1, #"Em atividade"

    #removendo quem não está fazendo o enem pra valer
    IN_TREINEIRO == 0, #"Não"

    #considerando apenas os que já concluíram
    #ou concluirão o ensino médio no mesmo ano desse enem 2021
    TP_ST_CONCLUSAO %in% c("Já concluí o Ensino Médio", "Estou
cursando e concluirei o Ensino Médio em 2021"),

    #removendo escolas federais e não respostas no tipo de escola
    #e garantindo que o tipo de escola condiz com a administração
```

```
    (TP_ESCOLA == "Pública" & TP_DEPENDENCIA_ADM_ESC %in%
c("Estadual", "Municipal")) |
      (TP_ESCOLA == "Privada" & TP_DEPENDENCIA_ADM_ESC ==
"Privada"),

    #apenas participante no ensino regular
    TP_ENSINO == "Ensino Regular"
  )
```

REMOVENDO VARIÁVEIS PARA ETAPA FUTURA

```{r}
enem2 = enem %>%
  select(NOTA_FINAL, TP_ESCOLA, TP_SEXO, TP_COR_RACA,
         REGIAO, CRITERIO,
         Q025,
         Q001,
         Q002,
         TP_LOCALIZACAO_ESC
  ) %>%
  rename(ESC = TP_ESCOLA,
         COR = TP_COR_RACA,
         REG = REGIAO,
         CRIT = CRITERIO,
         NET = Q025,
         SEX = TP_SEXO,
         NOTA = NOTA_FINAL,
         LOC = TP_LOCALIZACAO_ESC
  )

enem3 = enem2 %>% select(-SEX, -LOC, -NET)

enem4 = enem3 %>% select(-Q001)
```

AED

Frequencia por tipo de escola

```{r}
### Frequencia por tipo de escola

enem %>%
  group_by(TP_DEPENDENCIA_ADM_ESC) %>%
  summarise(n = n()) %>%
  mutate(freq = round(100*n / sum(n), 2))

enem %>%
  group_by(TP_ESCOLA) %>%
  summarise(n = n()) %>%
  mutate(freq = round(100*n / sum(n), 2))
```

Nota média por competência, por tipo da escola

```{r}
### Nota média geral, por tipo da escola
enem %>%
  group_by(TP_ESCOLA) %>%
  summarise(NOTA_FINAL = mean(NOTA_FINAL), PORTUGUES =
mean(NU_NOTA_LC), MATEMATICA = mean(NU_NOTA_MT), CN =
mean(NU_NOTA_CN), CH = mean(NU_NOTA_CH), REDACAO =
mean(NU_NOTA_REDACAO)) %>%
  melt("TP_ESCOLA")%>%
  group_by(variable, TP_ESCOLA) %>%
  ggplot(aes(x = variable, y = value, group = TP_ESCOLA, fill =
TP_ESCOLA)) +
  geom_bar(stat = "identity", position = position_dodge(0.9)) +
  geom_text(aes(label = round(..y.., 2)), position =
position_dodge(.9),
            fontface = "bold", vjust = 0) +
  ggtitle(NULL) +
  scale_y_continuous(limits = c(0, 1000), name = "Nota Média") +
  # scale_x_discrete(name = "Competência") +
  theme_bw() +
  theme(legend.position = c(.5, .95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.background = element_rect(color = "transparent", fill
= "transparent")) +
  # theme(legend.position = "none") +
  labs(x = "Competência")
```

Porcentagem da renda por tipo de escola

```{r}
### Porcentagem da renda por tipo de escola
enem %>%
  ggplot(aes(x = CRITERIO, group = TP_ESCOLA, fill = TP_ESCOLA)) +
  geom_bar(aes(y = ..prop.., fill = TP_ESCOLA), stat = "count",
position = 'dodge') +
  geom_text(aes(label = scales::percent(..count../tapply(..count..,
..x.. ,sum)[..x..], accuracy = .01), y = ..prop..),
            stat = "count",
            position = position_dodge(.9),
            fontface = "bold",
            vjust = 0) +
  scale_y_continuous(labels = scales::percent) +
  theme_bw() +
  theme(legend.position = c(.5, .95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        # legend.background = element_rect(color = "transparent",
fill = "transparent")
        ) +
  # guides(color = guide_legend(nrow = 1)) +
  labs(x = "CRITERIO", y = "prop") #+
# labs(fill = "Tipo de escola")
```

Nota por região IBGE, por tipo de escola

```{r}
### Nota por região IBGE, por tipo de escola

enem %>%
  group_by(REGIAO, TP_ESCOLA) %>%
  summarise(NOTA = mean(NOTA_FINAL)) %>%
  ggplot(aes(x = REGIAO, y = NOTA, fill = TP_ESCOLA)) +
  geom_bar(stat = "identity", position = "dodge") +
  theme_bw() +
  scale_y_continuous(limits = c(0, 1000), name = "Nota Média") +
  theme(legend.position = c(.5, .95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.background = element_rect(color = "transparent", fill
= "transparent")) +
  # guides(color = guide_legend(nrow = 1)) +
  labs(x = "REGIAO") +
  # ggtitle("Nota média por tipo de escola e região IBGE") +
  geom_text(aes(label = round(NOTA, 2)), position =
position_dodge(.9),
            fontface = "bold", vjust = 0)
```

Nota por critério, por tipo de escola

```{r}
### Nota por critério brasil, por tipo de escola

enem %>%
  group_by(TP_ESCOLA, CRITERIO) %>%
  summarise(NOTA = mean(NOTA_FINAL)) %>%
  ggplot(aes(x = TP_ESCOLA, y = NOTA, fill = CRITERIO)) +
  geom_bar(stat = "identity", position = "dodge") +
  theme_bw() +
  scale_y_continuous(limits = c(0, 1000), name = "Nota Média") +
  theme(legend.position = c(.5, .95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.background = element_rect(color = "transparent", fill
= "transparent")) +
  # guides(color = guide_legend(nrow = 1)) +
  labs(x = "REGIAO") +
  # ggtitle("Nota média por tipo de escola e região IBGE") +
  geom_text(aes(label = round(NOTA, 2)), position =
position_dodge(.9),
            fontface = "bold", vjust = 0)
```

Critério por região e nota por critério por região

```{r}
### a. Critério por Região v1
```

```
enem %>%
  ggplot(aes(x = REGIAO, group = CRITERIO, fill = CRITERIO)) +
  geom_bar(aes(y = ..prop.., fill = CRITERIO), stat = "count",
position = "dodge") +
  geom_text(aes(label = scales::percent(..prop.., accuracy = .01), y
= ..prop..),
            stat = "count",
            position = position_dodge(.95), vjust = 0,
            fontface = "bold") +
  ggtitle("Critério por Região") +
  scale_y_continuous(labels = scales::percent) +
  theme_bw() +
  theme(legend.position = c(.5, .95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.background = element_rect(color = "transparent", fill
= "transparent")) +
  # guides(color = guide_legend(nrow = 1)) +
  labs(x = "REGIAO", y = "prop por Classe", title = NULL) #+
# labs(fill = "Tipo de escola")

### a. Critério por região v2

enem %>%
  ggplot(aes(x = REGIAO, group = CRITERIO, fill = CRITERIO)) +
  geom_bar(aes(y = ..count../tapply(..count.., ..x.. ,sum)[..x..]),
stat = "count", position = "dodge") +
  geom_text(aes(label = scales::percent(..count../tapply(..count..,
..x.. ,sum)[..x..], accuracy = .01), y = ..count../tapply(..count..,
..x.. ,sum)[..x..]),
            stat = "count",
            position = position_dodge(.95),
            fontface = "bold",
            vjust = 0) +
  ggtitle("a. Proporção de Critério Brasil por Região IBGE") +
  scale_y_continuous(labels = scales::percent) +
  theme_bw() +
  theme(legend.position = c(.5, .95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.background = element_rect(color = "transparent", fill
= "transparent")) +
  # guides(color = guide_legend(nrow = 1)) +
  labs(x = "REGIAO", y = "prop") #+
# labs(fill = "Tipo de escola")
```

```{r}
### b. Nota por Critério Brasil, por região

enem %>%
  group_by(REGIAO, CRITERIO) %>%
  summarise(NOTA = mean(NOTA_FINAL)) %>%
  ggplot(aes(x = REGIAO, y = NOTA, fill = CRITERIO)) +
  geom_bar(stat = "identity", position = "dodge") +
  # scale_x_discrete(labels = function(x) str_wrap(x, width = 10)) +
```

```
  # facet_wrap(~ TP_ESCOLA) +
  theme_bw() +
  # scale_x_discrete(name = "") +
  scale_y_continuous(limits = c(0, 1000), name = "Nota Média") +
  theme(legend.position = c(.5, .95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.background = element_rect(color = "transparent", fill
= "transparent")) +
  # guides(color = guide_legend(nrow = 1)) +
  labs(x = "REGIAO") +
  ggtitle("b. Nota média por Critério Brasil") +
  geom_text(aes(label = round(NOTA, 2)), position =
position_dodge(.9),
            fontface = "bold", vjust = 0)
```

Nota por Critério Brasil, por tipo de escola

```{r}
### Nota por Critério Brasil, por tipo de escola
enem %>%
  group_by(CRITERIO, TP_ESCOLA) %>%
  summarise(NOTA = mean(NOTA_FINAL)) %>%
  ggplot(aes(x = CRITERIO, y = NOTA, fill = TP_ESCOLA)) +
  geom_bar(stat = "identity", position = "dodge") +
  # scale_x_discrete(labels = function(x) str_wrap(x, width = 10)) +
  # facet_wrap(~ TP_ESCOLA) +
  theme_bw() +
  scale_y_continuous(limits = c(0, 1000), name = "Nota Média") +
  theme(legend.position = c(.5, .95),
        legend.title = element_blank(),
        legend.direction = "horizontal",
        legend.background = element_rect(color = "transparent", fill
= "transparent")) +
  # theme(legend.position = "none") +
  labs(x = "CRITERIO") +
  # ggtitle("Nota média por tipo de escola e Critério Brasil") +
  geom_text(aes(label = round(NOTA, 2)), position =
position_dodge(.9),
            fontface = "bold", vjust = 0)
```

gráfico coroplético

```{r}
### gráfico coroplético

cidades = read_municipality() %>% mutate(code_muni =
as.factor(code_muni))
estados = read_state() %>% mutate(CO_UF_ESC = code_state)
regioes = read_region() %>% mutate(REGIAO = factor(name_region))
```

por estado

```{r}
### por estado
mapabr1 = merge(estados, enem %>%
                  group_by(CO_UF_ESC) %>%
                  summarise(NOTA = mean(NOTA_FINAL)), by =
"CO_UF_ESC", all = T) %>%
  ggplot(aes(geometry = geometry)) +

  geom_sf(aes(fill = NOTA),
          color = "transparent", size = 0.0
  ) +
  # facet_wrap(~TP_ESCOLA) +
  scale_fill_gradient(low = "red", high = "green",
                      na.value = "white", name = "Nota média") +
  theme_bw() +
  # scale_fill_distiller(palette = "Reds", limits=c(0, 1000),
  # name="Code_muni") +
  geom_text(aes(label = abbrev_state),
            stat = "sf_coordinates",
            # fontface = "bold"
  ) +
  theme(panel.grid = element_line(colour = "transparent"),
        panel.background = element_blank(), axis.text =
element_blank(),
        axis.ticks = element_blank()) +
  labs(x = element_blank(), y = element_blank()) +
  ggtitle("a. Nota média por estado")
```

por regiao

```{r}
### por regiao
mapabr2 = merge(regioes %>% select(-c(name_region, code_region)),
enem %>%
                  group_by(REGIAO) %>%
                  summarise(NOTA = mean(NOTA_FINAL)), by = "REGIAO",
all = T) %>%
  ggplot(aes(geometry = geometry)) +
  geom_sf(aes(fill = NOTA),
          color = "transparent", size = 0.0) +
  scale_fill_gradient(low = "red", high = "green",
                      na.value = "white", name = "Nota média") +
  theme_bw() +
  geom_text(aes(label = REGIAO),
            stat = "sf_coordinates",
            # fontface = "bold"
  ) +
  theme(panel.grid = element_line(colour = "transparent"),
        panel.background = element_blank(), axis.text =
element_blank(),
        axis.ticks = element_blank()) +
  labs(x = element_blank(), y = element_blank()) +
  ggtitle("b. Nota média por região")
```

```
gridExtra::grid.arrange(mapabr1, mapabr2, ncol = 2)
```

nota por estado

```{r}
### nota por estado

enem %>% group_by(SG_UF_ESC) %>% summarise(mean(NOTA_FINAL)) %>%
View()
```

TESTES DE MÉDIA

```{r}
dms = function(x1, x2, a = .05)
{
  teste = aov(x1 ~ x2)
  aux = sum(1/summary(droplevels(x2)))
  QMRes = sum(teste$residuals^2)/teste$df.residual
  dms = stats::qtukey(p = (1-a),
                      df = teste$df.residual,
                      nmeans = length(levels(droplevels(x2)))) *
    sqrt((QMRes/2) * (1/aux))
  return(dms)
}
```

Nota por tipo de escola

```{r}
#### Nota por tipo de escola

enem %>% group_by(TP_ESCOLA) %>%
  summarise(mean(NOTA_FINAL), median(NOTA_FINAL), sd(NOTA_FINAL))

teste = aov(NOTA_FINAL ~ TP_ESCOLA, data = enem)
summary(teste)
TukeyHSD(teste)

enem %>% group_by(TP_ESCOLA) %>%
  ggplot(aes(x = reorder(TP_ESCOLA, NOTA_FINAL, FUN = median,
decreasing = T), y = NOTA_FINAL, fill = reorder(TP_ESCOLA,
NOTA_FINAL, FUN = median, decreasing = T))) +
  geom_boxplot() +
  theme_bw() +
  labs(x = "TP_ESCOLA") +
  theme(legend.position = "none")
```

Nota por cor/raça

```{r}
#### Nota por cor/raça
enem %>% group_by(TP_COR_RACA) %>%
  summarise(mean(NOTA_FINAL), median(NOTA_FINAL), sd(NOTA_FINAL))
```

```
teste = aov(NOTA_FINAL ~ TP_COR_RACA, data = enem)
summary(teste)
TukeyHSD(teste)

enem %>% group_by(TP_COR_RACA) %>%
  ggplot(aes(x = reorder(TP_COR_RACA, NOTA_FINAL, FUN = median,
decreasing = T), y = NOTA_FINAL, fill = reorder(TP_COR_RACA,
NOTA_FINAL, FUN = median, decreasing = T))) +
  geom_boxplot() +
  theme_bw() +
  labs(x = "TP_COR_RACA") +
  theme(legend.position = "none")

## agrupando cor/raça != branca
enem %>%
  mutate(COR2 = ifelse(TP_COR_RACA != "Branca", "Não branca",
"Branca")) %>%
  group_by(COR2) %>%
  ggplot(aes(x = reorder(COR2, NOTA_FINAL, FUN = median, decreasing
= T), y = NOTA_FINAL, fill = reorder(COR2, NOTA_FINAL, FUN = median,
decreasing = T))) +
  geom_boxplot() +
  theme_bw() +
  labs(x = "COR2") +
  theme(legend.position = "none")
```

Nota por sexo

```{r}
#### Nota por sexo
enem %>% group_by(TP_SEXO) %>%
  summarise(mean(NOTA_FINAL), median(NOTA_FINAL), sd(NOTA_FINAL))

teste = aov(NOTA_FINAL ~ TP_SEXO, data = enem)
summary(teste)
TukeyHSD(teste)

enem %>% group_by(TP_SEXO) %>%
  ggplot(aes(x = TP_SEXO, y = NOTA_FINAL, fill = TP_SEXO)) +
  geom_boxplot() +
  theme_bw() +
  theme(legend.position = "none") +
  scale_fill_manual(values = c("#00BFC4", "#F8766D"))
```

Nota por região

```{r}
#### Nota por região
enem %>% group_by(REGIAO)  %>%
  summarise(mean(NOTA_FINAL), median(NOTA_FINAL), sd(NOTA_FINAL))

teste = aov(NOTA_FINAL ~ REGIAO, data = enem)
summary(teste)
```

```
TukeyHSD(teste)

enem %>% group_by(REGIAO) %>%
  ggplot(aes(x = reorder(REGIAO, NOTA_FINAL, FUN = median,
decreasing = T), y = NOTA_FINAL, fill = reorder(REGIAO, NOTA_FINAL,
FUN = median, decreasing = T))) +
  geom_boxplot() +
  theme_bw() +
  labs(x = "REGIAO") +
  theme(legend.position = "none")
```

Nota por Critério Brasil

```{r}
#### Nota por Critério Brasil

enem %>% group_by(CRITERIO) %>%
  summarise(mean(NOTA_FINAL), median(NOTA_FINAL), sd(NOTA_FINAL))

teste = aov(NOTA_FINAL ~ CRITERIO, data = enem)
summary(teste)
TukeyHSD(teste)

enem %>% group_by(CRITERIO) %>%
  ggplot(aes(x = CRITERIO, y = NOTA_FINAL, fill = CRITERIO)) +
  geom_boxplot() +
  theme_bw() +
  theme(legend.position = "none")
```

Nota por acesso à internet

```{r}
#### Nota por acesso à internet

enem %>% group_by(Q025)  %>%
  summarise(mean(NOTA_FINAL), median(NOTA_FINAL), sd(NOTA_FINAL))

teste = aov(NOTA_FINAL ~ Q025, data = enem)
summary(teste)
TukeyHSD(teste)

enem %>% group_by(Q025) %>%
  ggplot(aes(x = reorder(Q025, NOTA_FINAL, FUN = median, decreasing
= T), y = NOTA_FINAL), fill = Q025) +
  geom_boxplot(aes(fill = Q025)) +
  theme_bw() +
  labs(x = "Q025") +
  scale_fill_manual(values = c("#00BFC4", "#F8766D")) +
  theme(legend.position = "none")
```

Nota por escolaridade do pai

```{r}
```

```
#### Nota por escolaridade do pai

enem %>% group_by(Q001)  %>%
  summarise(mean(NOTA_FINAL), median(NOTA_FINAL), sd(NOTA_FINAL))

teste = aov(NOTA_FINAL ~ Q001, data = enem)
summary(teste)
TukeyHSD(teste)

enem %>% group_by(Q001) %>%
  ggplot(aes(x = Q001, y = NOTA_FINAL, fill = Q001)) +
  geom_boxplot() +
  theme_bw() +
  theme(legend.position = "none")
```

Nota por escolaridade da mãe

```{r}
#### Nota por escolaridade da mãe

enem %>% group_by(Q002)  %>%
  summarise(mean(NOTA_FINAL), median(NOTA_FINAL), sd(NOTA_FINAL))

teste = aov(NOTA_FINAL ~ Q001, data = enem)
summary(teste)
TukeyHSD(teste)

enem %>% group_by(Q002) %>%
  ggplot(aes(x = Q002, y = NOTA_FINAL, fill = Q002)) +
  geom_boxplot() +
  labs(x = "Q002") +
  theme_bw() +
  theme(legend.position = "none")
```

DAGs AUTOMÁTICO

```{r}
mygraph <- function(pcgraph){
  g <- as.bn(pcgraph, check.cycles = F)
  graphviz.plot(g, shape = "ellipse")
}
```

```{r}
#### matrix pra fixar relação
aux = matrix(data = F, ncol = 10, nrow = 10)
aux[2, 6] = aux[6, 2] = T
colnames(aux) = rownames(aux) = colnames(as.data.frame(enem2))
```

PC

padrão

```{r}
### pc
pcalg_fit_mix1 <- pc(suffStat = as.data.frame(enem2),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(as.data.frame(enem2)),
u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T)
mygraph(pcalg_fit_mix1)
```

pc fixando renda-escola

```{r}
### pc fixando renda-escola
pcalg_fit_mix2 = pc(suffStat = as.data.frame(enem2), indepTest =
mixCItest,
                     alpha = 0.05,
                     labels = colnames(as.data.frame(enem2)),
u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T,  solve.confl = T,
                     fixedEdges = aux)
mygraph(pcalg_fit_mix2)
```

pc forçando renda-escola, q001-nota e q002-nota

```{r}
### pc forçando renda-escola, q001-nota e q002-nota
aux3 = aux
aux3[8:9, 1] = aux3[1, 8:9] = T

pcalg_fit3_mix1 = pc(suffStat = as.data.frame(enem2),
                      indepTest = mixCItest, alpha = 0.05,
                      labels = colnames(enem2), u2pd="relaxed",
                      skel.method = "stable.fast",
                      maj.rule = T, solve.confl = T,
                      fixedEdges = aux3
)
mygraph(pcalg_fit3_mix1)
```

pc fixando renda-escola e q001-nota

```{r}
#### pc fixando renda-escola e q001-nota
aux4 = aux
aux4[8, 1] = aux4[1, 8] = T

pcalg_fit3_mix2 = pc(suffStat = as.data.frame(enem2),
                      indepTest = mixCItest, alpha = 0.05,
                      labels = colnames(enem2), u2pd="relaxed",
                      skel.method = "stable.fast",
                      maj.rule = T, solve.confl = T,
                      fixedEdges = aux4
```

```
)
mygraph(pcalg_fit3_mix2)
```

pc fixando renda-escola e q002-nota

```{r}
#### pc fixando renda-escola e q002-nota
aux5 = aux
aux5[9,1] = aux5[1,9] = T

pcalg_fit3_mix3 = pc(suffStat = as.data.frame(enem2),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem2), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     fixedEdges = aux5
)
mygraph(pcalg_fit3_mix3)
```

pc fixando q001-nota e q002-nota

```{r}
#### pc fixando q001-nota e q002-nota
aux6 = aux
aux6[,] = F
aux6[8:9, 1] = aux6[1,8:9] = T

pcalg_fit3_mix4 = pc(suffStat = as.data.frame(enem2),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem2), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     fixedEdges = aux6
)
mygraph(pcalg_fit3_mix4)
```

pc fixandoq001-nota

```{r}
aux7 = aux
aux7[,] = F
aux7[1,8] = aux7[8,1] = T

#### pc fixandoq001-nota
pcalg_fit3_mix5 = pc(suffStat = as.data.frame(enem2),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem2), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     fixedEdges = aux7
)
mygraph(pcalg_fit3_mix5)
```

pc fixando q002-nota

```{r}
aux8 = aux
aux8[,] = F
aux8[1,9] = aux8[9,1] = T

#### pc fixando q002-nota
pcalg_fit3_mix6 = pc(suffStat = as.data.frame(enem2),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem2), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     fixedEdges = aux8
)
mygraph(pcalg_fit3_mix6)
```

REMOVENDO SEX, LOC, NET

```{r}
auxB = aux[-c(3, 7, 10), -c(3, 7, 10)] #### removendo variaveis da
matrix, fixando renda-escola
```

pc removendo sex loc net

```{r}
### pc removendo sex loc net
pcalg_fit2_mix1 <- pc(suffStat = as.data.frame(enem3),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem3), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     # fixedEdges = aux
)
mygraph(pcalg2_fit_mix1)
```

pc removendo sex loc net

```{r}
### pc removendo sex loc net
pcalg_fit4_mix1 = pc(suffStat = as.data.frame(enem3),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem3), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     # fixedEdges = auxB
)
mygraph(pcalg_fit4_mix1)
```

pc removendo sex loc net, fixando renda-escola

```{r}
### pc removendo sex loc net, fixando renda-escola
# auxB
pcalg_fit4_mix3 = pc(suffStat = as.data.frame(enem3),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem3), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     fixedEdges = auxB
)
mygraph(pcalg_fit4_mix3)
```

pc removendo sex loc net, fixando q001-nota

```{r}
### pc removendo sex loc net, fixando q001-nota
auxB2 = auxB
auxB2[,] = F
auxB2[1,6] = auxB2[6,1] = T

pcalg_fit4_mix5 = pc(suffStat = as.data.frame(enem3),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem3), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     fixedEdges = auxB2
)
mygraph(pcalg_fit4_mix5)
```

pc removendo sex loc net, fixando q002-nota

```{r}
auxB3 = auxB
auxB3[,] = F
auxB3[1,7] = auxB3[7,1] = T
```

```{r}
### pc removendo sex loc net, fixando q002-nota
pcalg_fit4_mix7 = pc(suffStat = as.data.frame(enem3),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem3), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     fixedEdges = auxB3
)
mygraph(pcalg_fit4_mix7)
```

pc removendo sex loc net, fixando renda-escola + q001-nota

```{r}
auxB4 = auxB
```

```
auxB4[1,6] = auxB4[6,1] = T
auxB4
```

```{r}
### pc removendo sex loc net, fixando renda-escola + q001-nota
pcalg_fit4_mix9 = pc(suffStat = as.data.frame(enem3),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem3), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     fixedEdges = auxB4
)
mygraph(pcalg_fit4_mix9)
```

pc removendo sex loc net, fixando renda-escola + q002-nota

```{r}
auxB5 = auxB
auxB5[1,7] = auxB5[7,1] = T
auxB5
```

```{r}
###pc removendo sex loc net, fixando renda-escola + q002-nota
auxB5 = auxB
auxB5[1,7] = auxB5[7,1] = T
auxB5

pcalg_fit4_mix11 = pc(suffStat = as.data.frame(enem3),
                      indepTest = mixCItest, alpha = 0.05,
                      labels = colnames(enem3), u2pd="relaxed",
                      skel.method = "stable.fast",
                      maj.rule = T, solve.confl = T,
                      fixedEdges = auxB5
)
mygraph(pcalg_fit4_mix11)
```

pc removendo sex loc net, fixando q001-nota + q002-nota

```{r}
auxB6 = auxB
auxB6[,] = F
auxB6[1,6:7] = auxB6[6:7,1] = T
auxB6
```

```{r}
### pc removendo sex loc net, fixando q001-nota + q002-nota

pcalg_fit4_mix13 = pc(suffStat = as.data.frame(enem3),
                      indepTest = mixCItest, alpha = 0.05,
                      labels = colnames(enem3), u2pd="relaxed",
                      skel.method = "stable.fast",
```

```
                      maj.rule = T, solve.confl = T,
                      fixedEdges = auxB6
)
mygraph(pcalg_fit4_mix13)
```

pc removendo sex loc net, fixando renda-escola + q001-nota + q002-nota

```{r}
auxB7 = auxB
auxB7[1,6:7] = auxB7[6:7,1] = T
auxB7
```

```{r}
#### pc removendo sex loc net, fixando renda-escola + q001-nota + q002-nota
pcalg_fit4_mix15 = pc(suffStat = as.data.frame(enem3),
                      indepTest = mixCItest, alpha = 0.05,
                      labels = colnames(enem3), u2pd="relaxed",
                      skel.method = "stable.fast",
                      maj.rule = T, solve.confl = T,
                      fixedEdges = auxB7
)
mygraph(pcalg_fit4_mix15)
```

TAMBÉM REMOVENDO ESCOLARIDADE DO PAI (q001)

```{r}
auxC = auxB[-6, -6]
```

```{r}
### pc tambem removendo q001
pcalg_fit5_mix1 = pc(suffStat = as.data.frame(enem4),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem4), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     # fixedEdges = auxB
)
mygraph(pcalg_fit5_mix1)
```

pc tambem removendo q001, fixando renda-escola

```{r}
### pc tambem removendo q001, fixando renda-escola
# auxC
pcalg_fit5_mix3 = pc(suffStat = as.data.frame(enem4),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem4), u2pd="relaxed",
                     skel.method = "stable.fast",
```

```
                    maj.rule = T, solve.confl = T,
                    fixedEdges = auxC
)
mygraph(pcalg_fit5_mix3)
```

pc tambem removendo q001, fixando q002-nota

```{r}
### pc tambem removendo q001, fixando q002-nota
# auxC
auxC2 = auxC
auxC2[,] = F
auxC2[1,6] = auxC2[6,1] = T
```

```{r}
### pc tambem removendo q001, fixando q002-nota
pcalg_fit5_mix5 = pc(suffStat = as.data.frame(enem4),
                    indepTest = mixCItest, alpha = 0.05,
                    labels = colnames(enem4), u2pd="relaxed",
                    skel.method = "stable.fast",
                    maj.rule = T, solve.confl = T,
                    fixedEdges = auxC2
)
mygraph(pcalg_fit5_mix5)
```

pc também removendo q001, fixando renda-escola + q002-nota

```{r}
### pc também removendo q001, fixando renda-escola + q002-nota
# auxC
auxC3 = auxC
auxC3[1,6] = auxC3[6,1] = T
auxC3
```

```{r}
### pc também removendo q001, fixando renda-escola + q002-nota
pcalg_fit5_mix7 = pc(suffStat = as.data.frame(enem4),
                    indepTest = mixCItest, alpha = 0.05,
                    labels = colnames(enem4), u2pd="relaxed",
                    skel.method = "stable.fast",
                    maj.rule = T, solve.confl = T,
                    fixedEdges = auxC3
)
mygraph(pcalg_fit5_mix7)
```

FCI

padrão

```{r}
```

```
### fci
pcalg_fit_mix3 <- fci(suffStat = as.data.frame(enem2), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem2)),
                      skel.method = "stable.fast",
                      maj.rule = T)

plot(pcalg_fit_mix3)
```

fci fixando renda-escola

```{r}
### fci fixando renda-escola
pcalg_fit_mix4 <- fci(suffStat = as.data.frame(enem2), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem2)),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      fixedEdges = aux)
plot(pcalg_fit_mix4)
```

fci fixando renda-escola e q002-nota

```{r}
# aux5
pcalg_fit3_mix9 = fci(suffStat = as.data.frame(enem2), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem2)),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      fixedEdges = aux5)
plot(pcalg_fit3_mix9)
```

fci fixando q001-nota

```{r}
#### fci fixando q001-nota
# aux7
pcalg_fit3_mix11 = fci(suffStat = as.data.frame(enem2), indepTest =
mixCItest,
                       alpha = 0.05,
                       labels = colnames(as.data.frame(enem2)),
                       skel.method = "stable.fast",
                       maj.rule = T,
                       fixedEdges = aux7)
plot(pcalg_fit3_mix11)
```

fci fixando renda-escola e q001-nota

```{r}
#### fci fixando renda-escola e q001-nota
aux4 = aux
aux4[8, 1] = aux4[1, 8] = T

pcalg_fit3_mix13 = fci(suffStat = as.data.frame(enem2), indepTest =
mixCItest,
                       alpha = 0.05,
                       labels = colnames(as.data.frame(enem2)),
                       skel.method = "stable.fast",
                       maj.rule = T,
                       fixedEdges = aux4)
plot(pcalg_fit3_mix13)
```

fci removendo sex loc net

```{r}
### fci removendo sex loc net
pcalg_fit4_mix2 = fci(suffStat = as.data.frame(enem3), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem3)),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      # fixedEdges = auxB
)
plot(pcalg_fit4_mix2)
```

fci removendo sex loc net, fixando renda-escola

```{r}
### fci removendo sex loc net, fixando renda-escola
# auxB
pcalg_fit4_mix4 = fci(suffStat = as.data.frame(enem3), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem3)),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      fixedEdges = auxB
)
plot(pcalg_fit4_mix4)
```

fci removendo sex loc net, fixando q001-nota

```{r}
### fci removendo sex loc net, fixando q001-nota
# auxB2
pcalg_fit4_mix6 = fci(suffStat = as.data.frame(enem3), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem3)),
                      skel.method = "stable.fast",
```

```
                      maj.rule = T,
                      fixedEdges = auxB2
)
plot(pcalg_fit4_mix6)
```

fci removendo sex loc net, fixando q002-nota

```{r}
### fci removendo sex loc net, fixando q002-nota
# auxB3
pcalg_fit4_mix8 = fci(suffStat = as.data.frame(enem3), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem3)),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      fixedEdges = auxB3
)
plot(pcalg_fit4_mix8)
```

fci removendo sex loc net, fixando renda-escola + q001-nota

```{r}
### fci removendo sex loc net, fixando renda-escola + q001-nota
# auxB4
pcalg_fit4_mix10 = fci(suffStat = as.data.frame(enem3), indepTest =
mixCItest,
                       alpha = 0.05,
                       labels = colnames(as.data.frame(enem3)),
                       skel.method = "stable.fast",
                       maj.rule = T,
                       fixedEdges = auxB4
)
plot(pcalg_fit4_mix10)
```

fci removendo sex loc net, fixando renda-escola + q001-nota

```{r}
### fci removendo sex loc net, fixando renda-escola + q001-nota
# auxB5
pcalg_fit4_mix12 = fci(suffStat = as.data.frame(enem3), indepTest =
mixCItest,
                       alpha = 0.05,
                       labels = colnames(as.data.frame(enem3)),
                       skel.method = "stable.fast",
                       maj.rule = T,
                       fixedEdges = auxB5
)
plot(pcalg_fit4_mix12)
```

fci removendo sex loc net, fixando q001-nota + q002-nota

```{r}
### fci removendo sex loc net, fixando q001-nota + q002-nota
# auxB6
pcalg_fit4_mix14 = fci(suffStat = as.data.frame(enem3), indepTest =
mixCItest,
                       alpha = 0.05,
                       labels = colnames(as.data.frame(enem3)),
                       skel.method = "stable.fast",
                       maj.rule = T,
                       fixedEdges = auxB6
)
plot(pcalg_fit4_mix14)
```

fci removendo sex loc net, fixando renda-escola + q001-nota + q002-nota

```{r}
### fci removendo sex loc net, fixando renda-escola + q001-nota +
q002-nota
# auxB7

pcalg_fit4_mix16 = fci(suffStat = as.data.frame(enem3), indepTest =
mixCItest,
                       alpha = 0.05,
                       labels = colnames(as.data.frame(enem3)),
                       skel.method = "stable.fast",
                       maj.rule = T,
                       fixedEdges = auxB7
)
plot(pcalg_fit4_mix16)
```

fci tambem removendo q001

```{r}
pcalg_fit5_mix2 = fci(suffStat = as.data.frame(enem4), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem4)),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      # fixedEdges = auxB
)
plot(pcalg_fit5_mix2)
```

fci tambem removendo q001, fixando renda-escola

```{r}
### fci tambem removendo q001, fixando renda-escola
# auxC
pcalg_fit5_mix4 = fci(suffStat = as.data.frame(enem4), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem4)),
```

```
                      skel.method = "stable.fast",
                      maj.rule = T,
                      fixedEdges = auxC
)
plot(pcalg_fit5_mix4)
```

fci tambem removendo q001, fixando q002-nota

```{r}
### fci tambem removendo q001, fixando q002-nota
pcalg_fit5_mix6 = fci(suffStat = as.data.frame(enem4), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem4)),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      fixedEdges = auxC2
)
plot(pcalg_fit5_mix6)
```

fci também removendo q001, fixando renda-escola + q002-nota

```{r}
### fci também removendo q001, fixando renda-escola + q002-nota
# auxC3
pcalg_fit5_mix8 = fci(suffStat = as.data.frame(enem4), indepTest =
mixCItest,
                      alpha = 0.05,
                      labels = colnames(as.data.frame(enem4)),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      fixedEdges = auxC3
)
plot(pcalg_fit5_mix8)
```

hill-climbing

padrão

```{r}
#### fixando (wL) / barrando (bl) relações para o hill-climbing

wl = matrix(c("CRIT", "ESC"),
            ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

wl1 = matrix(c("CRIT", "NOTA"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

wl2 = matrix(c("Q001", "NOTA"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))
```

```
wl3 = matrix(c("Q002", "NOTA"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

wl4 = matrix(c("Q001", "NOTA", "Q002", "NOTA"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

wl5 = matrix(c("CRIT", "ESC", "Q001", "NOTA"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

wl6 = matrix(c("CRIT", "ESC", "Q002", "NOTA"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

wl7 = matrix(c("CRIT", "ESC", "Q001", "NOTA", "Q002", "NOTA"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

bl1 = matrix(c("CRIT", "SEX", "NET", "SEX", "ESC", "SEX", "COR",
"SEX",
               "Q001", "SEX", "Q002", "SEX", "ESC", "CRIT", "CRIT",
"COR",
               "ESC", "Q001", "ESC", "Q002", "SEX", "COR", "SEX",
"CRIT",
               "NET", "LOC", "REG", "SEX", "LOC", "SEX"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

bnlearn_hc = bnlearn::hc(enem2,
                         whitelist = wl2,
                         blacklist = bl1
)
graphviz.plot(bnlearn_hc, shape = "ellipse"
              # , highlight = list(arcs = wl)
)
```

hill-climbing removendo sex loc net

```{r}
bl2 = matrix(c("ESC", "Q001", "ESC", "Q002", "ESC", "COR", "ESC",
"CRIT",
               "CRIT", "COR"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

bnlearn_hc = bnlearn::hc(enem3,
                         # whitelist = wl2,
                         blacklist = bl2
)
graphviz.plot(bnlearn_hc, shape = "ellipse"
              # , highlight = list(arcs = wl)
)
```

```
```

hill-climbing também removendo q001

```{r}
bl3 = matrix(c("ESC", "Q002", "ESC", "COR", "ESC", "CRIT",
               "CRIT", "COR"),
             ncol = 2, byrow = TRUE, dimnames = list(NULL, c("from",
"to")))

bnlearn_hc = bnlearn::hc(enem4,
                         # whitelist = wl2,
                         blacklist = bl3
)
graphviz.plot(bnlearn_hc, shape = "ellipse"
              # , highlight = list(arcs = wl)
)
```

AGRUPANDO Q002-B COM Q002-H

pc agrupando escolaridade da mãe

```{r}
### pc agrupando escolaridade da mãe
pcalg_fit6_mix1 = pc(suffStat = as.data.frame(enem2 %>% mutate(Q102
= fct_collapse(Q002, B = c("B", "H"))) %>% select(-c(Q001, Q002,
SEX, LOC, NET))),
                     indepTest = mixCItest, alpha = 0.05,
                     labels = colnames(enem2 %>% mutate(Q102 =
fct_collapse(Q002, B = c("B", "H"))) %>% select(-c(Q001, Q002, SEX,
LOC, NET))), u2pd="relaxed",
                     skel.method = "stable.fast",
                     maj.rule = T, solve.confl = T,
                     # fixedEdges = auxC3
)
mygraph(pcalg_fit6_mix1)
```

fci agrupando escolaridade da mãe

```{r}
### fci agrupando escolaridade da mãe
pcalg_fit6_mix2 = fci(suffStat = as.data.frame(enem2 %>% mutate(Q102
= fct_collapse(Q002, B = c("B", "H"))) %>% select(-c(Q001, Q002,
SEX, LOC, NET))), indepTest = mixCItest,
                      alpha = 0.05,
                      labels = colnames(enem2 %>% mutate(Q102 =
fct_collapse(Q002, B = c("B", "H"))) %>% select(-c(Q001, Q002, SEX,
LOC, NET))),
                      skel.method = "stable.fast",
                      maj.rule = T,
                      # fixedEdges = auxC3
)
plot(pcalg_fit6_mix2)
```

ANEXO A – Quadros de pontuação para o Critério Brasil

Tabela 10 – Pontuação para o Critério Brasil - Variáveis

	Quantidade				
	0	**1**	**2**	**3**	**4 ou +**
Banheiros	0	3	7	10	14
Empregados domésticos	0	3	7	10	13
Automóveis	0	3	5	8	11
Microcomputador	0	3	6	8	11
Lava-louça	0	3	6	6	6
Geladeira	0	2	3	5	5
Freezer	0	2	4	6	6
Lava-roupa	0	2	4	6	6
DVD	0	1	3	4	6
Micro-ondas	0	2	4	4	4
Motocicleta	0	1	3	3	3
Secadora de roupas	0	2	2	2	2

Tabela 11 – Pontuação para Critério Brasil - escolaridade do chefe de família

Grau de instrução do chefe da família	**Pontuação**
Analfabeto / Fundamental I incompleto	0
Fundamental I completo / Fundamental II incompleto	1
Fundamental II completo / Médio incompleto	2
Médio completo / Superior incompleto	7
Superior completo	8