

Relatório: modelagem e descrição dos resultados

Convex-Lab*

Grupo de desenvolvimento do modelo

Enviado em 27 de Setembro de 2020

Apresentação do Relatório

Neste relatório uma estratégia composta por um modelo matemático de otimização linear e um procedimento iterativo é apresentada para solução do problema da ONG Matemática em Movimento. O modelo matemático foi implementado em Python e os resultados obtidos não revelaram inconsistências no modelo ou código, mesmo após testes com diferentes dados de entrada. Adicionalmente aos requisitos apresentados pela ONG, outros opcionais foram incluídos no modelo, os quais podem ser utilizados ou não, de acordo com as necessidades da ONG. Com relação à estrutura do relatório, inicialmente a estratégia de solução é brevemente apresentada (1), em seguida o modelo matemático (2) que compõe a estratégia é apresentado em termos de: índices, variáveis de decisão, parâmetros (2.1 e 2.2), função objetivo (2.3) e restrições (2.4). Em seguida detalhes relacionados à implementação do modelo são delineados (3), como: versões da linguagem de programação e bibliotecas utilizadas (3.1), pré-processamentos necessários (3.2) e estrutura do código (3.3). Por fim os resultados (4) do modelo são comentados, assim como os testes conduzidos para averiguação dos mesmos.

1 Estratégia para solução do problema

A estratégia apresentada para solução do problema é composta por duas fases. Na primeira, resolve-se um modelo de otimização linear para alocar os alunos de continuidade e preencher as vagas das turmas pré-existentes. Para isso, o modelo minimiza o número de turmas e o número de vagas livres nessas turmas. Nessa primeira fase, o modelo garante que novas turmas são criadas apenas se forem necessárias para alocar os alunos de continuidade, de acordo com a função objetivo. Além disso, o modelo aloca alunos de formulário apenas para preencher essas vagas, de acordo com a ordem de prioridade de data de inscrição.

Em seguida, na segunda fase um procedimento iterativo é responsável por criar novas turmas para os alunos de formulário, de acordo com o orçamento da ONG. Os alunos de formulário que já foram inscritos na primeira fase são desconsiderados na segunda fase. Cada iteração da segunda fase realiza os seguintes procedimentos: (i) verifica se a ONG ainda tem orçamento disponível e escolhe a série e escola a serem abertas de acordo com os critérios do desafio. Nessa etapa, consideram-se os critérios de desempate mais recentes propostos no desafio; (ii) preenche uma nova turma na série e escola, ambas escolhidas pelos candidatos de formulário, de acordo com a prioridade na inscrição. A segunda fase realiza iterações para abertura de novas turmas enquanto o orçamento máximo da ONG permitir e enquanto existirem alunos de formulário para serem alocados em turmas.

* Contato:

Camila Pereira dos Santos Tautenhain - camila.p.san@gmail.com

Dalton Garcia Borges de Souza - dalgarbor@gmail.com

2 Modelo Matemático

Tem-se definidos seguintes índices, que são:

- n_c : Número de alunos de continuidade;
- n_f : Número de alunos de formulário;
- u : Limite superior do número de turmas possíveis;
- m_s : Número de séries;
- m_e : Número de escolas;
- U_t : Conjunto de turmas já existentes.

As variáveis de decisão do modelo são as seguintes:

- xc_{it} : 1 se o aluno de continuidade i é alocado na turma t , 0 caso contrário;
- xf_{it} : 1 se o aluno de formulário i é alocado na turma t , 0 caso contrário;
- y_{ts} : 1 se a turma t é da série s , 0 caso contrário;
- z_{te} : 1 se a turma t vai ser alocada na escola e , 0 caso contrário.
- vt_t : 1 se a turma t vai ser aberta, 0 caso contrário.

Adicionalmente, tem-se uma variável de folga:

- vf_t : variável de folga que indica quantas vagas não foram preenchidas na turma t .

Os parâmetros do modelo são divididos em parâmetro identificados pelo modelo (2.1) e parâmetros de entrada (2.2).

2.1 Parâmetros identificados pelo modelo

Os parâmetros criados para o funcionamento do modelo são:

- $p_{ri} = 1$: se o aluno de continuidade i reprovou, 0 caso contrário;
- $p_{tx_t} = 1$: se a turma t já existia, 0 caso contrário;
- p_{ts_t} : série da turma $t \in U_t$ pre-existente;
- p_{te_t} : escola da turma $t \in U_t$ pre-existente;
- p_{oi_i} : vetor que indica o ranqueamento do aluno, de acordo com a ordem de inscrição dos alunos de formulário;
- p_{at_i} : turma na qual o aluno de continuidade estudava;
- p_{as_i} : série na qual o aluno de continuidade precisa ser alocado;
- p_{fs_i} : série na qual o aluno de formulário precisa ser alocado;
- p_{ae_i} : escola na qual o aluno de continuidade vai estudar;
- p_{fe_i} : escola na qual o aluno de formulário vai estudar.

2.2 Parâmetros de entrada

Os parâmetros passados pela ONG são:

- q_ma : quantidade máxima de alunos por turma, nomeado previamente como "qtd_max_alunos";
- q_pp : quantidade de professores pedagógicos por turma, previamente como "qtd_professores_pedagogico";
- q_pa : quantidade de professores ACD por turma, nomeado previamente como "qtd_professores_acd";
- lc : limite de custo da ONG no período, nomeado previamente como "limite_custo";
- cp : custo por professor, nomeado previamente como "custo_professor";
- ca : custo por aluno, nomeado previamente como "custo_aluno".

2.3 Função objetivo

O objetivo do modelo é minimizar as variáveis de folga (vf_t), que indicam quantas vagas sobram em cada turma, e o número de turmas abertas. Dessa forma, maximiza-se a alocação de alunos nas turmas. Além disso, após a alocação dos alunos de continuidade, a função objetivo deve priorizar os alunos de formulário, que são ranqueados em ordem crescente de acordo com a ordem de inscrição. Abordagens de Maximização da razão entre o número total de alunos e o número total de turmas poderiam levar a não-linearidades, por isso foram evitadas.

A função objetivo é dada por:

$$\min \sum_{t=1}^u vf_t + \sum_{t=1}^u vt_t + \sum_{i=1}^n \sum_{t=1}^u 0.1 p_{oi} x_{fi} \quad (1)$$

2.4 Restrições

R1: Se alunos i e j estavam na mesma turma, então eles tem que continuar, desde que nenhum deles tenha sido reprovado.

$$x_{it} = x_{jt}, \forall t \in \{1, 2, \dots, u\} \text{ e } \forall i, j \in \{1, 2, \dots, n_c\} \text{ tais que } p_{ri} = p_{rj} = 0 \text{ e } p_{at_i} = p_{at_j} \quad (2)$$

R2: Cada aluno deve ser alocado na serie apropriada, determinada de acordo com as regras.

$$xc_{it} \leq y_{ts}, s = p_{as_i}, \quad \forall i \in \{1, 2, \dots, n_c\}, \forall t \in \{1, 2, \dots, u\} \quad (3)$$

$$xf_{it} \leq y_{ts}, s = p_{fs_i}, \quad \forall i \in \{1, 2, \dots, n_f\}, \forall t \in \{1, 2, \dots, u\} \quad (4)$$

R3: Cada aluno i só pode ser alocado na turma t se ela existir na escola e que ele escolheu.

$$xc_{it} \leq z_{te}, e = p_{ae_i}, \quad \forall i \in \{1, 2, \dots, n_c\}, \forall t \in \{1, 2, \dots, u\} \quad (5)$$

$$xf_{it} \leq z_{te}, e = p_{fe_i}, \quad \forall i \in \{1, 2, \dots, n_f\}, \forall t \in \{1, 2, \dots, u\} \quad (6)$$

R4: Turmas pré-existentes já tem a série e a escola definidas.

$$\forall y_{ts} = 1, s = p_{ts_t}, \quad \forall t \in Ut \quad (7)$$

$$\forall z_{te} = 1, e = p_{te_t}, \quad \forall t \in Ut \quad (8)$$

R5: Alunos de continuidade tem que obrigatoriamente ser alocados em uma turma.

$$\sum_{t=1}^u xc_{it} = 1, \quad \forall i \in \{1, 2, \dots, n_c\} \quad (9)$$

R6: Cada aluno de formulário é alocado em no máximo uma turma.

$$\sum_{t=1}^u xf_{it} \leq 1, \quad \forall i \in \{1, 2, \dots, n_f\} \quad (10)$$

R7: Cada turma pertence a uma única serie e escola:

$$\sum_{s=1}^{m_s} y_{ts} = 1, \quad \forall t \in \{1, 2, \dots, u\} \quad (11)$$

$$\sum_{e=1}^{m_e} z_{te} = 1, \quad \forall t \in \{1, 2, \dots, u\} \quad (12)$$

R8: Número máximo de alunos por turma. Nessa restrição vf_t é uma variável de folga que indica quantas vagas sobram na turma t , se a turma t for aberta.

$$\left(\sum_{i=1}^{n_c} xc_{it} + \sum_{i=1}^{n_f} xf_{it} \right) + vf_t = qtd_max_alunos \, vt_t \quad t \in \{1, 2, \dots, u\} \quad (13)$$

R9: Limite de custo máximo da ONG.

$$\sum_{t=1}^u \left((q_pp + q_pa)cp \, vt_t + ca \left(\sum_{i=1}^{n_c} xc_{it} + \sum_{i=1}^{n_f} xf_{it} \right) \right) \leq lc \quad (14)$$

3 Implementação

3.1 Linguagem e bibliotecas

Após formulação, o modelo matemático foi implementado em Python (versão 3.6.9) e utilizando-se apenas bibliotecas gratuitas no processo. São elas:

- pandas: utilizada para análise de dados e manipulação de tabelas, ou *dataframes*. Versão utilizada: 1.0.5
- numpy: utilizada para manipulação de vetores e matrizes. Versão utilizada: 1.18.5
- sqlite3: utilizada na criação de bancos de dados em disco. Versão utilizada: 2.6.0
- scipy: utilizada para cálculos matemáticos e otimizações. Versão utilizada: 1.4.1.
- PuLP: API utilizada na resolução do modelo, por meio do emprego de solvers externos. No caso deste trabalho, utilizou-se o solver CBC. Versão utilizada do PuLP: 2.3. Versão utilizada do CBC: 2.9.0.

Caso tenha versões mais atuais dessas bibliotecas e esteja experimentando problemas na execução do código, considere reduzir as versões às mencionadas acima. Adicionalmente a este relatório e ao código, um arquivo do tipo "README" é disponibilizado para facilitar a utilização da solução apresentada.

3.2 Pré-processamentos necessários

Após importação dos dados em Python, alguns pré-processamentos foram realizados nos dados, a fim de ajustá-los ao modelo. São eles:

- Na tabela "Aluno", fez-se um ajuste da série pretendida pelo aluno. Por padrão aumenta-se uma série. Também verifica-se a série que o aluno quer cursar está ativa. Alunos que cursariam séries inativas são desconsiderados no planejamento.
- Para os "Alunos de continuidade", desconsiderou-se os alunos que não continuariam na ONG, por escolha ou formatura. Também ajustou-se o ano real que os alunos cursariam.
- Para os "Alunos de formulário", identificou-se a próxima série dos alunos, quando o ano informado era menor que o de planejamento.
- Na tabela "Série", deixa-se apenas as séries ativas da ONG, assim como a última série ativa da ONG é determinada.

3.3 Estrutura do código

A fim de aumentar a modularidade e a reusabilidade da solução desenvolvida, grande parte do código é destinado à definição de funções de auxílio. Essas funções são responsáveis pela importação e importação dos dados do SQL, pelos pré-processamentos abordados na sub-seção anterior (3.2), pela declaração das estruturas do modelo matemático (2) e pelo procedimento iterativo para abertura de novas turmas que compõe a estratégia proposta. Por fim, todas as informações são incorporadas ao código principal (*Main*), que retorna as saídas do modelo, salvando-as no banco de dados original.

4 Resultados

Executando-se o código para o banco de dados fornecido, o solver foi capaz de encontrar soluções factíveis para o problema. Das 10 turmas pré-existentes, as turmas do 9º ano e uma do 2º ano de foram fechadas e outras 12 turmas foram abertas. Dessas, uma nova turma de 9º ano na zona oeste foi aberta, agora apenas para para um aluno. Nem todo orçamento foi utilizado, já que não haviam alunos de formulário suficientes para requisitar todo orçamento passado.

A nomeação das turmas também não mostrou contradições. Turmas fechadas e que tiveram similares abertas, receberam o mesmo nome. Nenhum nome duplicado foi constatado, assim como erros nas séries.

A fim de se assegurar o desempenho do modelo em situações diferentes daquela retratada pelo atual banco de dados, testou-se os seguintes cenários isoladamente:

1. O limite de custo foi ajustado para R\$ 15.000;
2. A possibilidade de se otimizar dentro do ano foi ativada;
3. A possibilidade de se abrir novas turmas foi desativada;
4. Os dez primeiros alunos de continuação foram tratados como reprovados;
5. Somente alunos de formulário foram considerados.
6. Somente alunos de formulário foram considerados e o custo foi ajustado para R\$ 5.000.

Em quatro dos seis cenários o modelo retornou resultados factíveis. As únicas exceções esperadas foram o primeiro e terceiro testes, já que a insuficiência do orçamento ou o impedimento de se abrir novas turmas impossibilitava a alocação dos alunos de continuidade. Sendo assim, o modelo retornou alertas em ambos os casos. Os banco de dados para cada cenário se encontram na pasta "Cenários", que segue em conjunto com este relatório.