

Usando Python e o Google Colab Para Ensinar Físico-Química

Leonardo Baptista

Universidade do Estado do Rio de Janeiro, Faculdade de Tecnologia, Departamento de Química e Ambiental (leobap@gmail.com)

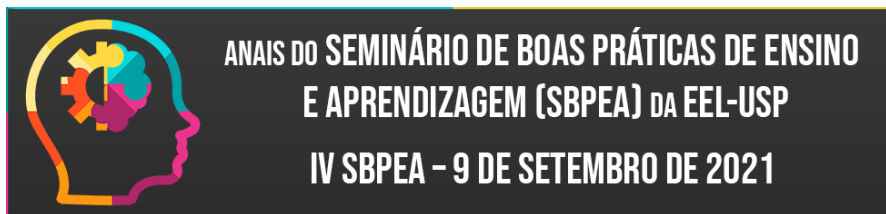
Resumo

A pandemia de COVID-19 trouxe a necessidade dos professores reinventarem seus métodos pedagógicos a fim de engajar os estudantes em aulas remotas. Encontros síncronos e assíncronos se tornaram rotina em cursos de graduação, e métodos de ensino baseados na internet se mostraram úteis no atual cenário. Por estes motivos, a plataforma Google Colab pode ser uma boa ferramenta para o ensino de físico-química. Adicionalmente, a plataforma pode ajudar os estudantes a desenvolverem suas habilidades no uso de ferramentas colaborativas e uso da linguagem Python para solução de problemas de físico-química com uso do conhecimento matemático e computacional adquiridos nas disciplinas básicas. Oito notebooks do Jupyter foram desenvolvidos e usados via Google Colab na disciplina de físico-química do curso de Engenharia Química da Faculdade de Tecnologia da UERJ, como ferramenta para resolução de exercícios propostos ao longo da disciplina. Estes notebooks fizeram parte das aulas assíncronas da disciplina e o retorno positivo por parte dos alunos indicou que a plataforma pode continuar a ser usada nos semestres seguintes.

Palavras-chave: Físico-Química, Métodos de Ensino Baseado na Internet, Termodinâmica, Equilíbrio, Python.

Abstract

The COVID-19 pandemic brought the need to reinvent teaching pedagogical methods to engage undergraduate students in remote classes. Synchronous and asynchronous



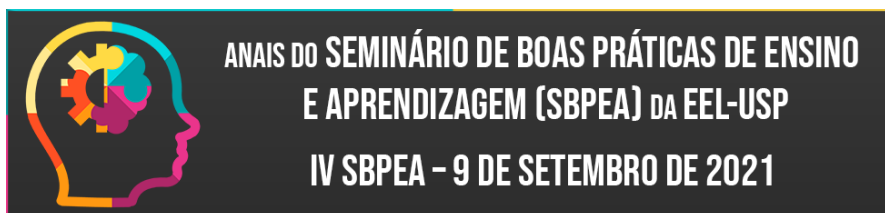
meetings become routine in the undergraduate courses, and web-based learning methods proved to be useful in the present situation. In this scenario, the Google Colab platform could be a good tool to teach physical-chemistry. Also, help the students develop different skills using collaborative tools, solve physical-chemistry problems using Python language, and apply mathematical knowledge learned in the basic courses. Eight Jupyter notebooks were developed and used, via Google Colab, in the physical chemistry course of the Technological Faculty of the Rio de Janeiro State University to participate in the asynchronous classes and help the students solve the assigned problems. The students' positive feedback pointed out that this initiative was well accepted and should be continued in the next semesters.

Keywords: Physical Chemistry, Web-Based Learning, Thermodynamics, Equilibrium, Python.

1. INTRODUÇÃO

Já é bem conhecido o fato de que a disciplina de físico-química é complexa e muitas vezes considerada abstrata devido ao conteúdo abordado e conhecimento matemático necessário. Durante o período da pandemia de COVID-19 foi observado um aumento na dificuldade das disciplinas da graduação devido à excepcionalidade do momento. E o mesmo ocorreu com a disciplina de físico-química (HWANG, 2020; PERETS *et al.*, 2020; SOARES *et al.*, 2020; WILD *et al.*, 2020). Devido às aulas remotas, os professores precisaram se esforçar mais para garantir o engajamento dos alunos e o desejado desempenho acadêmico nas disciplinas (EMENIKE *et al.*, 2020; LAWRIE, 2021; RANGA, 2020).

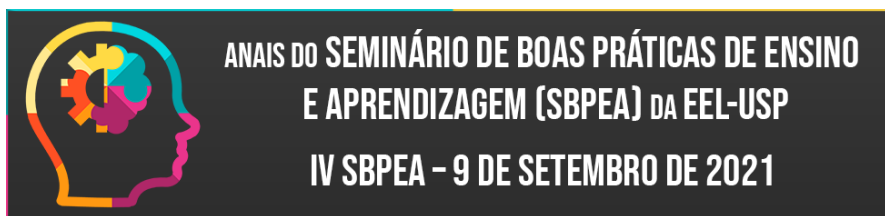
Muitos problemas da disciplina envolvem o cálculo de derivadas, integrais, resolução de equações diferenciais ordinárias e ajuste de modelos matemáticos a dados experimentais. Todos estes procedimentos podem ser feitos com o auxílio de softwares disponíveis na internet. Por exemplo, regressões lineares podem ser feitas utilizando o Microsoft Excel ou o Libre Office Calc, normalmente as primeiras escolhas da maior dos alunos. No entanto, estes pacotes não são adequados para cálculos avançados e tem seu uso limitado na atividade profissional de químicos e engenheiros químicos. Por



estes motivos, o curso de físico-química pode ser a porta de entrada para o uso de recursos computacionais necessários para resolução de problemas matemáticos avançados, tratamento e análise de dados.

Python é uma linguagem computacional simples, que contém uma série de bibliotecas disponíveis, como Numpy (Numpy, 2021), SciPy (SciPy.org, 2021), Matplotlib (Matplotlib, 2021), além de outras que permitem realizar simulações, ajuste de modelos matemáticos, resolução de equações diferenciais ordinárias e construção de gráficos. O leitor interessado em conhecer mais sobre os recursos e capacidades do Python é incentivado a acessar alguns artigos disponíveis (DAHLGREN, 2018; JARVIS *et al.*, 2006; MENKE, 2020; SRNEC; UPADHYAY; MADURA, 2017; WEISS, 2017). Tendo em vista a facilidade da linguagem Python, o Google criou a plataforma Colab (Google Colab, 2021), na qual é possível escrever e rodar códigos em Python além de efetuar cálculos usando Tensor Flow. Tudo isso usando a própria infraestrutura em nuvem do Google. A plataforma não requer download, instalação ou qualquer tipo de configuração do Python e suas bibliotecas no computador pessoal dos novos usuários. Estas facilidades ajudam os professores no uso de notebooks do Jupyter (KLUYVER *et al.*, 2016) para escrever *scripts* em Python que serão usados em suas aulas remotas, visto que não demanda conhecimento prévio por parte dos estudantes da instalação e configuração do Python. O Google Colab é uma plataforma inclusiva, pois pode ser usada gratuitamente e só requer um dispositivo conectado à internet, sendo possível escrever e executar códigos em Python a partir de smartphones e tablets.

O principal objetivo do presente trabalho é mostrar que é possível ensinar físico-química e introduzir os estudantes à linguagem Python, via plataforma Google Colab, durante o curso. Adicionalmente, mostrar como a linguagem de programação escolhida é uma ferramenta útil, além de fácil, para resolver problemas de físico-química. Avaliar os recursos da plataforma do Google no monitoramento, em tempo real, do progresso dos estudantes e como esta pode ajudar a resolver de forma rápida as dificuldades com os problemas propostos. Além disso, mostrar que é possível transformar o curso de físico-química em um curso interdisciplinar que incentive os estudantes a conectarem e



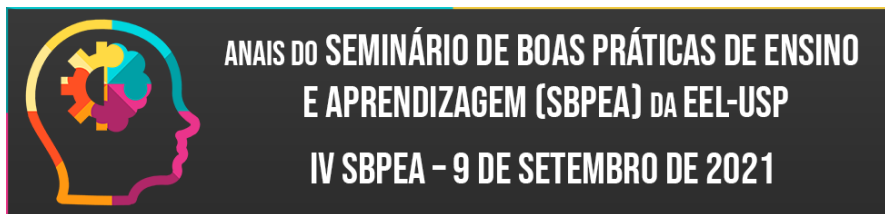
aplicarem os conhecimentos adquirido nos cursos introdutórios de cálculo e computação com os problemas da físico-química.

2. FUNDAMENTAÇÃO PEDAGÓGICA

Todos os scripts foram escritos como notebooks do Jupyter (KLUYVER *et al.*, 2016) no formato iPython (PÉREZ; GRANGER, 2007) e disponibilizados aos alunos pela plataforma Colab, via compartilhamento do link dos notebooks no Google Sala de Aula usado para disciplina. De acordo com as permissões dadas ao notebook, múltiplos usuários podem editar e executar o código Python. O ambiente é naturalmente colaborativo, ideal para os períodos de ensino remoto e distanciamento social. O uso do Google Colab é uma ferramenta adicional para o ensino de físico-química e engajamento dos alunos na disciplina. Além de ajudar os estudantes a conectarem conceitos interdisciplinares e a ganhar habilidades necessárias para sua vida profissional (HILL *et al.*, 2019; KURNIAWAN *et al.*, 2019).

É permitido que os estudantes salvem estes notebooks para uso pessoal, além de ser permitido a edição e adição de comentários, via balões laterais, os quais permitem o monitoramento da atividade dos estudantes e suas dúvidas. Os notebooks foram escritos com o objetivo de resolverem problemas propostos na literatura básica do curso (ATKINS; PAULA, 2018; LEVINE, 2009; METZ, 1990; NEVERS, 2012), usando as bibliotecas Numpy, SciPy e Matplotlib. Foi assumido que os estudantes não possuem conhecimento prévio da linguagem Python, por esta razão, os *scripts* foram escritos da forma mais simples possíveis e todas as instruções foram comentadas. Já é sabido que o formato do curso pode causar um impacto negativo nos alunos, por esta razão foi escolhida uma ferramenta que não adicionaria uma dificuldade extra ao aprendizado de físico-química.

Como os notebooks fizeram parte do conteúdo assíncrono da disciplina, um vídeo tutorial foi disponibilizado junto aos notebooks, a fim de explicar o código e o método usado para resolução dos problemas propostos. Um texto em linguagem Markdown (Markdown Guide, 2021) foi adicionado aos notebooks para descrever os problemas e adicionar uma explicação das soluções propostas.

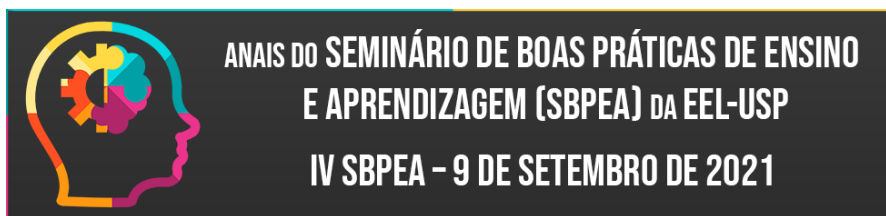


Ao longo do semestre foi passado aos alunos um trabalho acadêmico com uma lista de problemas que necessitavam o uso de ferramentas computacionais para resolvê-los. No entanto, como conhecimento de Python não é um pré-requisito da disciplina, os estudantes ficaram livres para escolher o software a ser usado na solução dos exercícios propostos.

3. ESTRUTURA E CONTEÚDO DOS SCRIPTS

O Google Colab é uma plataforma na qual é possível escrever notebooks do Jupyter, usando o formato iPython, contendo elementos interativos e que pode ser compartilhado por diversos usuários. O ambiente é ideal para visualização de dados, cálculos científicos e processamento de dados. Textos adicionais podem ser incluídos em células específicas usando o formato Markdown, o qual pode ser usado para introduzir problemas e explicar a solução proposta (Figura 1). Os notebooks podem ser compartilhados entre os usuários sem a necessidade de um repositório adicional, tornando fácil a disponibilidade dos códigos e monitoramento do progresso dos estudantes. Os notebooks são salvos automaticamente na conta do usuário no Google Drive, não havendo a necessidade de se baixar os arquivos para o computador pessoal. Usuários podem adicionar comentários a cada célula do notebook para discutir suas dúvidas ou problemas relacionados com as tarefas. Esta ferramenta facilita a comunicação entre estudantes e professores e a comunicação entre colegas de trabalho. É um meio seguro para os estudantes comunicarem dúvidas ao professor referentes a resolução de exercícios, além de permitir o acompanhamento das dificuldades na resolução de exercícios. Nos balões de comentários existe um botão denominado “Resolvido”, que ao ser clicado indica que a dúvida ou problema enfrentado foi solucionado. Se necessário, a questão pode ser reaberta para discussões e correções adicionais, como ilustrado na Figura 1, número 5.

A plataforma tem outros recursos que permitem os professores monitorarem o progresso e atividades dos estudantes. Como mostrado na Figura 1 (nº 1), é possível visualizar qual foi a última vez que o notebook foi editado e qual usuário executou o código (nº 4). Estas características são ideais para métodos de aprendizagem baseados na *Web* e o ensino remoto.

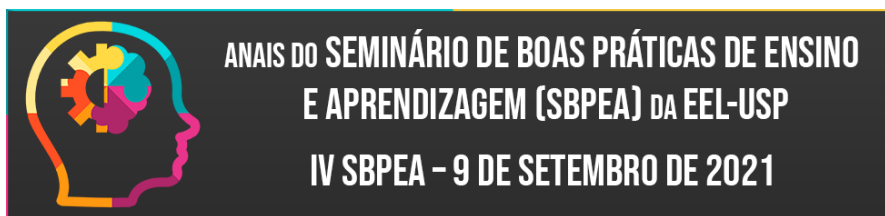


É possível instalar outras bibliotecas que não estão no Colab por padrão usando comandos nativos do Linux. O ambiente suporta Pandas, Jupyter *widgets*, pode acessar dados em planilhas do Google ou arquivos do Google Drive. É uma plataforma adequada para cálculos científicos, permitindo que usuário que faça seus cálculos sem instalação e configuração de pacotes computacionais. Estas características tornam a plataforma ideal para testes, novos usuários que desejam usar ferramentas computacionais e aqueles que têm pouca familiaridade com ferramentas computacionais. Os notebooks estão descritos nas seções abaixo e disponíveis para download em um repositório do Git-Hub (apesar da versão encontrada no Git-Hub estar em inglês, a versão usada nas aulas de físico-química está inteiramente em português).

The screenshot displays a Google Colab notebook interface. At the top, the browser tabs and address bar are visible. The notebook title is "Exercícios-Fis-Qui-Gases.ipynb". The main content area is divided into sections labeled (1) through (5):

- (1) A code cell containing the title "Exercício 1)".
- (2) A text cell describing the problem: "Para 1.0000 mol de N_2 a 0.00°C os seguintes volumes são observados em função da pressão:" followed by a table of pressure (p/atm) and volume (V/cm³) data points.
- (3) A code cell containing Python code to define vectors, calculate pV/nT, and perform a polynomial fit to find the gas constant R.
- (4) A text cell showing the output of the code: "[-3.61522973e-05 8.20593996e-02]".
- (5) A comment box on the right side of the notebook, showing a user profile for "Leonardo Baptista" and a comment input field.

Figura 1: Trecho de um *script* do Python escrito com o Google Colab. 1) Data na qual o *script* foi editado pela última vez; 2) Células de texto, escritas em formato Markdown, usadas para descrição dos problemas e soluções propostas; 3) Célula usada para escrita do código Python; 4) Caixa de diálogo mostrando quando o código foi executado pela última vez e o nome do usuário que o executou; 5) A coluna da direita mostra uma célula de comentário na qual alunos podem descrever suas dúvidas e receber instruções por parte do professor.



a. Notebook 1: Gás Ideal

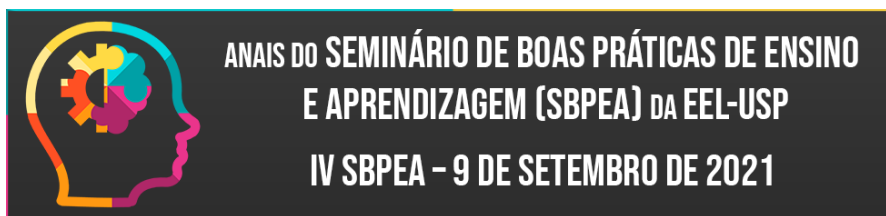
O primeiro notebook inclui: como fazer uma regressão linear a partir de dados experimentais fornecidos, o procedimento para cálculo da constante universal dos gases, o método para determinar a massa molecular de um gás e como a escala de temperatura do gás ideal pode ser obtida. Os gráficos foram construídos com o auxílio do Matplotlib com o objetivo de inspecionar visualmente regressão linear feita e introduzir os estudantes a construção de gráficos a partir de dados experimentais e modelos matemáticos.

b. Notebook 2: Primeira Lei

O segundo notebook tem o objetivo de resolver problemas relacionados ao cálculo da variação de entalpia e trabalho de expansão após aquecimento de um sistema. Os estudantes podem se familiarizar com a formatação de textos usando a linguagem Markdown, que foi utilizada para escrever o enunciado dos problemas, equações e soluções propostas para cada exercício. Neste notebook, o trabalho é calculado considerando que o gás se comporta primeiramente de forma ideal e em seguida como um gás de van der Waals. Finalmente, é pedido aos estudantes que calculem o trabalho de expansão de um gás que obedece uma expansão do tipo virial e de um gás que obedece a equação de estado de Redlich-Kwong.

c. Notebook 3: Entropia

Este notebook foi escrito com o objetivo de calcular a entropia da terceira lei de uma determinada substância. Inicialmente os estudantes são apresentados ao procedimento de ajuste de funções polinomiais a dados de capacidade calorífica em função da temperatura. Posteriormente gráficos contendo os valores experimentais e as funções ajustadas foram construídos. Após a escolha de um determinado intervalo de temperaturas, o cálculo da entropia foi feito pela integração numérica da função capacidade calorífica usando as funções implementadas na biblioteca SciPy. Problemas relacionados ao cálculo da entropia no limite de baixa temperatura foram discutidos e a Lei de Debye aT^3 é apresentada como uma maneira de se obter a entropia a baixas temperaturas.



d. **Notebook 4: Equilíbrio Químico**

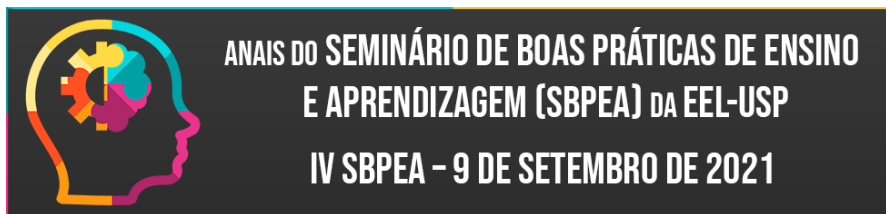
Neste notebook foi apresentada a rotina `scipy.optimize.fsolve` e como esta pode ser usada para solução de problemas envolvendo equilíbrio químico. Esta rotina encontra a raiz de uma função ou raízes de um sistema de equações lineares definidos pelo usuário na forma $f(x) = 0$. Problemas de equilíbrio químico simples foram sugeridos e resolvidos por meio desta rotina. O notebook mostra como escrever um problema de equilíbrio químico em linguagem Python e como resolvê-lo usando a biblioteca SciPy. Foi proposto o cálculo do mínimo da energia livre de Gibbs, que corresponde a condição de equilíbrio, usando a rotina `scipy.optimize.fsolve`. O notebook apresenta a solução de problemas que envolvem o efeito da temperatura e pressão na composição de equilíbrio, além de incluir correções de não idealidade das espécies gasosas na constante de equilíbrio.

e. **Notebook 5: Equilíbrio de fases de substâncias puras**

O código deste notebook foi escrito com o objetivo de resolver problemas envolvendo o equilíbrio de fases de substâncias puras. O notebook mostra como a entalpia de transição de fase da água pode ser calculada a partir de uma regressão linear da equação de Clausius-Clayperon a dados experimentais de equilíbrio líquido-vapor, assumindo que entalpia é constante no intervalo de temperatura considerado. Este notebook também foi usado para discutir a estabilidade das fases e como a pressão de vapor é afetada pela temperatura.

f. **Notebook 6: Gases Reais**

O código escrito neste notebook mostra como calcular o coeficiente de fugacidade de um gás via integração numérica, como calcular o segundo coeficiente do virial e a temperatura de Boyle de um gás. O aluno é apresentado a duas metodologias distintas para integração numérica: a regra do trapézio e o método de Simpson. A ideia é mostrar que ambos os métodos são simples serem usados em Python e que a escolha dependerá principalmente do problema a ser resolvido. A temperatura de Boyle foi estimada pelo ajuste de uma equação de segundo grau seguido de cálculo das raízes da equação obtida.



Como mais de uma raiz real pode ser obtida, foi feita uma discussão de qual raiz contém a solução desejada.

g. Notebook 7: Equilíbrio de fase de misturas

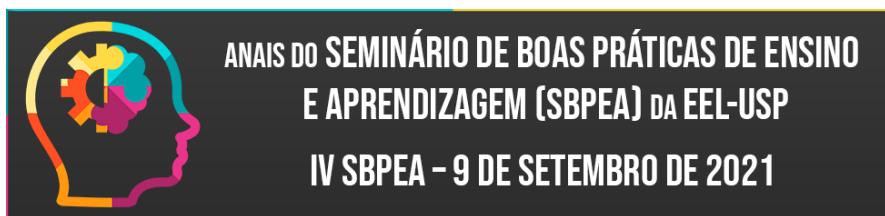
Este notebook apresenta ao estudante problemas mais complexos envolvendo o equilíbrio de fases de misturas. Também foram discutidos neste notebook os conceitos de solução ideal e solução idealmente diluída. Foram propostos problemas nos quais um ou mais processos de equilíbrio ocorrem simultaneamente e as soluções foram discutidas caso-a-caso. Por exemplo, um dos problemas envolve três processos simultâneos de equilíbrio e é mostrado como a regra de Cramer pode ser usada em Python para encontrar a solução de um sistema linear de equações.

h. Notebook 8: Equilíbrio Líquido-Vapor

Este código foi escrito para ensinar aos alunos como resolver sistemas de equações não-lineares e problemas de equilíbrio líquido-vapor considerando diferentes equações de estado. Foram mostradas as equações que devem ser satisfeitas quando uma determinada substância líquida está em equilíbrio com seu vapor, assumindo que o gás obedeça a equação de estado de Redlich-Kwong. Foi mostrado como o código pode ser usado para calcular a pressão de vapor e volume de saturação da fase gasosa e líquida do propano, usando a rotina `scipy.optimize.least_squares`. Depois que é mostrada a solução do exercício, foi pedido aos alunos que repitam os cálculos usando as equações de estado de van der Waals e Soave-Redlich-Kwong.

4. ATIVIDADES DA DISCIPLINA

O curso de físico-química é ministrado em 2 encontros semanais, nos quais são aplicados testes semanais referentes ao conteúdo abordado nas últimas aulas e um trabalho individual que deve entregue ao fim do semestre. O trabalho individual cobre todo o conteúdo da disciplina e contém uma lista de problemas que necessitam de ferramentas computacionais para serem resolvidos. Os estudantes têm a liberdade de usarem qualquer software, e como era de se esperar, a maioria dos estudantes escolhe por utilizar planilhas eletrônicas como a ferramenta computacional de seu trabalho. No entanto, devido às limitações das planilhas eletrônicas, os estudantes que escolhem este



tipo de programa tendem a resolver parte das atividades manualmente ou com o auxílio de um segundo software. No entanto, a partir do uso dos notebooks em Python como material suplementar do curso, uma mudança de cenário foi verificada. Dos 20 inscritos na disciplina no primeiro semestre de 2020, 55% dos estudantes usou integralmente ou parcialmente a linguagem Python via Google Colab para resolver os exercícios propostos. Enquanto que no segundo semestre de 2020, com exceção de um estudante, toda a turma resolveu os exercícios propostos em Python via Google Colab.

5. AVALIAÇÃO

Um questionário foi aplicado aos alunos da disciplina ao fim do primeiro semestre de 2020, e foi respondido por 15 alunos. A avaliação das respostas indicou que os estudantes consideraram como positivo o uso do Google Colab nas aulas de físico-química. Foi apontado pelos estudantes que a facilidade de uso e a não necessidade de download e instalação como as melhores características da plataforma. As quatro principais questões referentes ao uso do Google Colab e linguagem Python estão descritas na Figura 2. Nove dos 15 estudantes consideraram os *scripts* em Python um bom material suplementar do curso. Mas nem todos os estudantes consideraram os *scripts* úteis para o aprendizado de Python ou físico-química. A completa aceitação da ferramenta por todos os estudantes não era esperada, visto que muitos estudantes têm dificuldades em programação e cálculo numérico. Por estas razões muitos ainda insistem em resolver problemas matemáticos complexos em planilhas eletrônicas por serem mais amigáveis e já conhecidas.

Do ponto de vista do professor, a possibilidade de conversar com os alunos via balões de comentários foi muito útil para correção dos exercícios e código escrito pelos alunos. O compartilhamento dos notebooks acelerou a comunicação entre professor e aluno, além de melhorar o retorno dos alunos quanto ao conhecimento adquirido. Com relação aos notebooks compartilhados, foi possível visualizar quem editou e executou os *scripts* pelos próprios relatórios gerados pela plataforma. Estas informações se tornaram uma maneira de seguir o progresso dos estudantes e o tempo gasto em cada exercício.

No atual momento não é possível afirmar que a modalidade *online*, mediada pelo Google Colab, tem resultados comparáveis com o formato tradicional, como já reportado para o caso de cursos de química inorgânica (NENNIG *et al.*, 2020). No entanto, é possível sugerir o Google Colab como um recurso de valor para o ensino de físico-química na modalidade *online*, devido ao retorno positivo dos alunos, simplicidade para uso e solução de exercícios propostos.

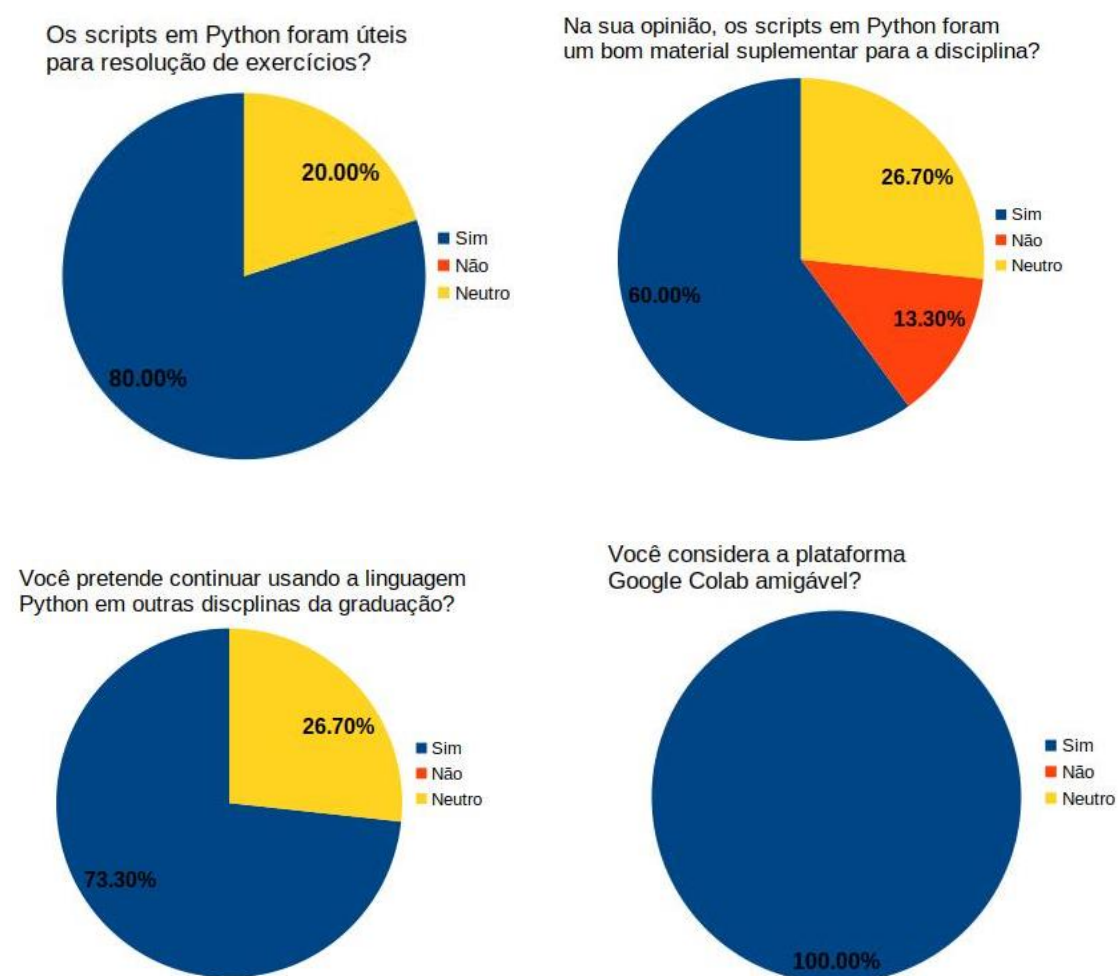
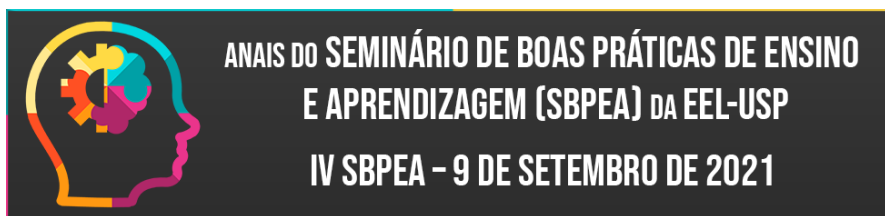


Figura 2: Questionário sobre o uso de scripts em Python via Google Colab aplicado ao fim do semestre de 2020/1. Quinze estudantes responderam os questionário.



6. CONSIDERAÇÕES FINAIS

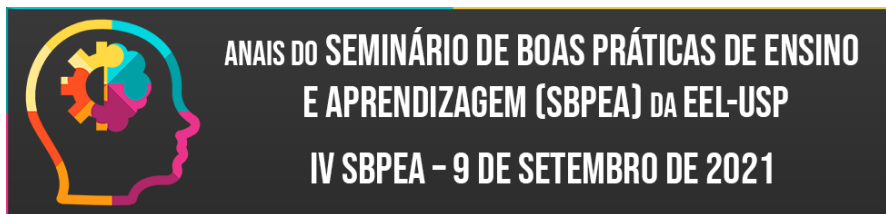
Oito notebooks do Jupyter foram escritos em linguagem Python e usados como material suplementar da disciplina de físico-química do curso de Engenharia Química da Faculdade de Tecnologia da UERJ. Estes notebooks foram apresentados aos alunos e tornados disponíveis via plataforma Google Colab. A plataforma foi usada como parte das aulas assíncronas da disciplina e escolhida por fornecer um ambiente para programação em Python sem a necessidade de instalação e configuração de pacotes e bibliotecas. Todos os estudantes que cursaram a disciplina consideraram a plataforma amigável e muitos a utilizaram para resolução de problemas propostos ao longo do curso. A plataforma é naturalmente multiusuário, algo que facilita a comunicação entre estudantes e professor, além ser uma ferramenta para *feedback* dos alunos no que tange os problemas propostos e as dificuldades relacionadas a solução destes. Ao fim dos dois semestres em que a plataforma foi usada, esta se mostrou um bom ambiente para trabalhos em grupo, proposição de exercícios e resolução de exercícios.

Os notebooks foram desenvolvidos utilizando poucos recursos da plataforma, entretanto, estas se mostraram muito úteis para o período de ensino remoto. Outros recursos mais avançados poderão ser explorados no futuro, como o uso de *dataframes* do Pandas, integração com as planilhas do Google e Jupyter *widgets*, a fim de aprimorar a metodologia de ensino usada no curso de físico-química.

A plataforma foi usada nos semestres letivos de 2020/1 e 2020/2 como parte do curso de físico-química, e devido ao retorno positivo por parte dos estudantes, continuará a ser utilizada em semestres posteriores. Possivelmente após o término da pandemia, o uso da plataforma e da linguagem Python se tornarão parte integral do curso de físico-química.

7. MATERIAL SUPLEMENTAR

Todos os códigos em Python citados neste trabalho estão disponíveis como notebooks do Jupyter no formato iPython no link <https://github.com/leobap01/physical-chemistry>. Todos os notebooks podem ser enviados diretamente para a plataforma Google Colab ou executados localmente, caso o usuário tenham instalado Python e suas bibliotecas.



8. REFERÊNCIAS

ATKINS, Peter; PAULA, Julio De. *Físico-química*. 10ª ed. Rio de Janeiro: LTC, 2018.

DAHLGREN, Björn. ChemPy: A package useful for chemistry written in Python.

Journal of Open Source Software, v. 3, n. 24, p. 565, 2018. Disponível em:

<<https://doi.org/10.21105/joss.00565>>. Acesso em: 21 jan. 2021.

EMENIKE, Mary E. *et al.* Leveraging undergraduate learning assistants to engage students during remote instruction: Strategies and lessons learned from four institutions.

Journal of Chemical Education, v. 97, n. 9, p. 2502–2511, 8 set. 2020. Disponível em:

<<https://pubs.acs.org/doi/abs/10.1021/acs.jchemed.0c00779>>. Acesso em: 5 jan. 2021.

Google Colab. Disponível em: <colab.research.google.com>. Acesso em: 21 jan. 2021.

HILL, Michelle A. *et al.* Undergraduate recognition of curriculum-related skill development and the skills employers are seeking. *Chemistry Education Research and Practice*, v. 20, n. 1, p. 68–84, 9 jan. 2019. Disponível em:

<<https://pubs.rsc.org/en/content/articlehtml/2019/rp/c8rp00105g>>. Acesso em: 21 jan. 2021.

HWANG, Candy S. Using Continuous Student Feedback to Course-Correct during COVID-19 for a Nonmajors Chemistry Course. *Journal of Chemical Education*, v. 97, n. 9, p. 3400–3405, 8 set. 2020. Disponível em:

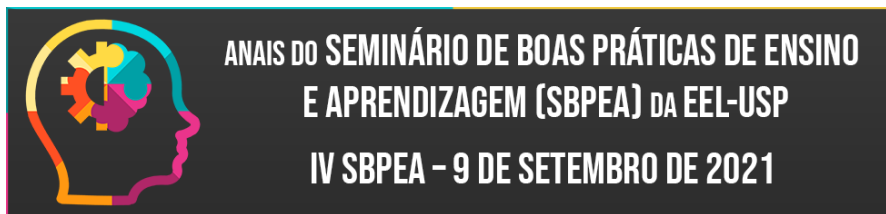
<<https://pubs.acs.org/doi/10.1021/acs.jchemed.0c00808>>. Acesso em: 4 jan. 2021.

JARVIS, R. M. *et al.* PYCHEM: a multivariate analysis package for python.

Bioinformatics, v. 22, n. 20, p. 2565–2566, 15 out. 2006. Disponível em:

<<https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btl416>>. Acesso em: 21 jan. 2021.

KLUYVER, Thomas *et al.* Jupyter Notebooks - a publishing format for reproducible computational workflows. 2016, Netherlands: IOS Press, 2016. p. 87–90. Disponível em: <<https://eprints.soton.ac.uk/403913/>>.



KURNIAWAN, Oka *et al.* Helping Students Connect Interdisciplinary Concepts and Skills in Physical Chemistry and Introductory Computing: Solving Schrödinger's Equation for the Hydrogen Atom. *Journal of Chemical Education*, v. 96, n. 10, p. 2202–2207, 2019.

LAWRIE, Gwendolyn. Chemistry education research and practice in diverse online learning environments: resilience, complexity and opportunity! *Chemistry Education Research and Practice*, v. 22, n. 1, p. 7, 14 jan. 2021. Disponível em: <<https://pubs.rsc.org/en/content/articlehtml/2021/rp/d0rp90013c>>. Acesso em: 21 jan. 2021.

LEVINE, Ira N. *Physical chemistry*. 6th. ed. New York: McGraw-Hill, 2009.

Markdown Guide. Disponível em: <<https://www.markdownguide.org/>>. Acesso em: 8 jan. 2021.

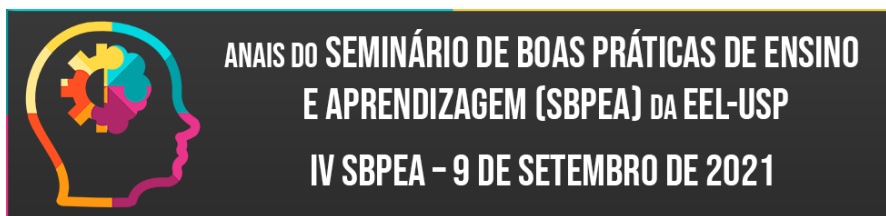
Matplotlib: Visualization with Python. Disponível em: <<https://matplotlib.org/>>. Acesso em: 21 jan. 2021.

MENKE, Erik J. Series of Jupyter Notebooks Using Python for an Analytical Chemistry Course. *Journal of Chemical Education*, v. 97, n. 10, p. 3899–3903, 2020. Disponível em: <[moz-extension://d2dd6a60-6a05-4dc8-96d4-2c4527e65fd9/enhanced-reader.html?openApp&pdf=https%3A%2F%2Fpubs.acs.org%2Fdoi%2Fpdf%2F10.1021%2Facs.jchemed.9b01131](https://pubs.acs.org/doi/pdf/10.1021/acs.jchemed.9b01131)>. Acesso em: 10 dez. 2020.

METZ, Clyde R. *Schaum's solved problems series - 2000 solved problems in physical-chemistry*. 1st ed: McGraw-Hill, Inc., 1990.

NENNIG, Hannah T. *et al.* Comparison of student attitudes and performance in an online and a face-to-face inorganic chemistry course. *Chemistry Education Research and Practice*, v. 21, n. 1, p. 168–177, 16 jan. 2020. Disponível em: <<https://pubs.rsc.org/en/content/articlehtml/2020/rp/c9rp00112c>>. Acesso em: 21 jan. 2021.

NEVERS, Noel De. *Physical and chemical equilibrium for chemical engineers*. 2nd. ed. Hoboken: John Wiley & Sons, Inc., 2012.



Numpy. Disponível em: <<https://numpy.org/>>. Acesso em: 21 jan. 2021.

PERETS, Ethan A. *et al.* Impact of the emergency transition to remote teaching on student engagement in a non-stem undergraduate chemistry course in the time of covid-19. *Journal of Chemical Education*, v. 97, n. 9, p. 2439–2447, 8 set. 2020. Disponível em: <<https://pubs.acs.org/doi/abs/10.1021/acs.jchemed.0c00879>>. Acesso em: 5 jan. 2021.

PÉREZ, Fernando; GRANGER, Brian E. iPython: a System for Interactive Scientific Computing. *Computing in Science and Engineering*, v. 9, n. 3, p. 21–29, 2007. Disponível em: <<https://ipython.org/>>.

RANGA, Jayashree S. Online Engagement of Commuter Students in a General Chemistry Course during COVID-19. *Journal of Chemical Education*, v. 97, n. 9, p. 2866–2870, 8 set. 2020. Disponível em: <<https://pubs.acs.org/doi/abs/10.1021/acs.jchemed.0c00633>>. Acesso em: 7 jan. 2021.

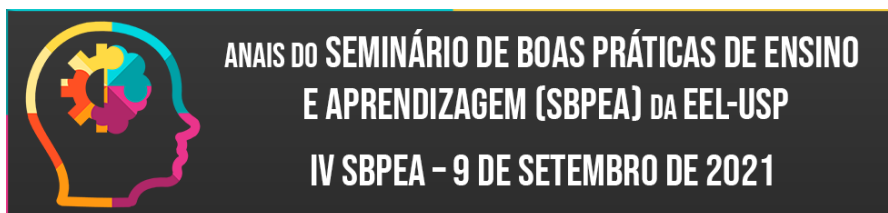
SciPy.org. Disponível em: <<https://www.scipy.org/>>. Acesso em: 21 jan. 2021.

SOARES, Ricardo *et al.* Online chemistry education challenges for rio de janeiro students during the covid-19 pandemic. *Journal of Chemical Education*, v. 97, n. 9, p. 3396–3399, 8 set. 2020. Disponível em: <<https://pubs.acs.org/doi/abs/10.1021/acs.jchemed.0c00775>>. Acesso em: 4 jan. 2021.

SRNEC, Matthew N.; UPADHYAY, Shiv; MADURA, Jeffry D. A Python Program for Solving Schrödinger's Equation in Undergraduate Physical Chemistry. *Journal of Chemical Education*, v. 94, n. 6, p. 813–815, 13 jun. 2017.

WEISS, Charles J. Scientific Computing for Chemists: An Undergraduate Course in Simulations, Data Processing, and Visualization. *Journal of Chemical Education*, v. 94, n. 5, p. 592–597, 9 maio 2017. Disponível em: <<https://pubs.acs.org/doi/pdf/10.1021/acs.jchemed.7b00078>>. Acesso em: 10 dez. 2020.

WILD, Duncan A. *et al.* Lessons Learned by Converting a First-Year Physical Chemistry Unit into an Online Course in 2 Weeks. *Journal of Chemical Education*, v. 97, n. 9, p. 2389–2392, 8 set. 2020. Disponível em:



<<https://dx.doi.org/10.1021/acs.jchemed.0c00678>>. Acesso em: 5 jan. 2021.