

**UNIVERSIDADE MUNICIPAL DE SÃO CAETANO DO SUL
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA
PROGRAMA DE PÓS-GRADUAÇÃO EM EDUCAÇÃO
MESTRADO PROFISSIONAL**

Alessandra Soares dos Santos

Profa. Dra. Adriana Barroso de Azevedo

**LINGUAGEM PYTHON NA SALA DE AULA: PRÁTICAS
INOVADORAS NO ENSINO DE PROBABILIDADE E ESTATÍSTICA
NO ENSINO MÉDIO**

**São Caetano do Sul - SP
2025**

**LINGUAGEM PYTHON NA SALA DE AULA: PRÁTICAS
INOVADORAS NO ENSINO DE PROBABILIDADE E ESTATÍSTICA
NO ENSINO MÉDIO**

Alessandra Soares Dos Santos

Profa. Dra. Adriana Barroso de Azevedo

1ª. Edição

Universidade Municipal de São Caetano do Sul - USCS

**São Caetano do Sul - SP
2025**

FICHA CATALOGRÁFICA

SANTOS, Alessandra Soares dos

LINGUAGEM PYTHON NA SALA DE AULA: PRÁTICAS INOVADORAS NO ENSINO DE PROBABILIDADE E ESTATÍSTICA NO ENSINO MÉDIO.

ISBN nº. 978-65-01-73185-8

Alessandra Soares dos Santos. – 2025.

Orientação: Profa. Dra. Adriana Barroso de Azevedo. Dissertação (mestrado) – USCS, Universidade Municipal de São Caetano do Sul, Programa de Mestrado em Educação, 2025.

1. Tecnologias digitais. 2. Matemática. 3. Probabilidade e Estatística. 4. Ensino Médio.

LISTA DE ABREVIATURAS E SIGLAS

BNCC	Base Nacional Comum Curricular
OCDE	Organização para a Cooperação e Desenvolvimento Econômico
PISA	Programa Internacional de Avaliação de Estudantes
Saeb	Sistema de Avaliação da Educação Básica
UFPI	Universidade Federal do Piauí
USCS	Universidade Municipal de São Caetano do Sul

SUMÁRIO

APRESENTAÇÃO	6
1 CONTEXTUALIZAÇÃO DO PRODUTO EDUCACIONAL.....	8
2 O OBJETO DO PRODUTO EDUCACIONAL	10
3 METODOLOGIA APLICADA NO PRODUTO EDUCACIONAL	11
4 SOBRE AS AUTORAS	14
5 CONSIDERAÇÕES FINAIS	15
REFERÊNCIAS	17
APÊNDICE A – FORMULÁRIO DE INSCRIÇÃO	21
APÊNDICE B – FORMULÁRIO DE FEEDBACK DO 1º. ENCONTRO	22
APÊNDICE C – FORMULÁRIO DE FEEDBACK DO 2º. ENCONTRO	23
APÊNDICE D – FORMULÁRIO DE FEEDBACK DO EVENTO.....	24
APÊNDICE E – MOMENTOS DA FORMAÇÃO	25
APÊNDICE F – CONTEÚDO PROGRAMÁTICO DO 1º ENCONTRO	31
APÊNDICE G – CONTEÚDO PROGRAMÁTICO DO 2º. ENCONTRO	44
APÊNDICE H – CONTEÚDO PROGRAMÁTICO DO 3º. ENCONTRO	60
APÊNDICE I – LISTA DE EXERCÍCIOS DO 1º. ENCONTRO.....	69
APÊNDICE J – LISTA DE EXERCÍCIOS DO 2º. ENCONTRO.....	76
APÊNDICE K – NARRATIVAS DOCENTES	88
ANEXO A – CERTIFICADO DO CURSO DE EXTENSÃO UNIVERSITÁRIA.....	90

APRESENTAÇÃO

Este produto educacional representa a materialização de uma jornada de pesquisa e prática no âmbito do mestrado em educação da USCS (Universidade Municipal de São Caetano do Sul), com o objetivo de oferecer uma resposta concreta e inovadora aos desafios do ensino da Matemática no Brasil. Acredita-se que a formação continuada de educadores, aliada à introdução de ferramentas tecnológicas relevantes como a programação, é essencial para empoderar professores, engajar estudantes e construir pontes entre o saber formal e as demandas de um mundo em constante transformação digital.

Lopes (2010, p. 35) argumenta que o letramento estatístico no Brasil “ainda é incipiente, pois os alunos chegam ao final da Educação Básica sem compreender conceitos básicos de probabilidade e estatística, muitas vezes por uma abordagem desconectada de contextos reais”.

Essa percepção reforçou a proposta de oferecer a formação em programação em Python para os conteúdos probabilísticos como uma ferramenta tecnológica alternativa para o ensino, corroborando com as ideias de Papert (1985) que afirma: “a programação computacional pode ser um veículo poderoso para o desenvolvimento de processos cognitivos e a construção de conhecimentos matemáticos”, refletindo a visão de Valente e Almeida (2021, p. 47).

Com isso, apresentamos o curso de extensão universitária “técnicas de programação sobre Probabilidade e Estatística para aula de Matemática do ensino médio”. Desenvolvido em parceria com a Universidade Federal do Piauí (UFPI), esse curso gratuito e homologado visa capacitar docentes da Educação Básica, fornecendo-lhes ferramentas didáticas inovadoras através da linguagem de programação Python para o ensino de Probabilidade e Estatística. Com uma carga horária total de 60 horas, distribuídas em 12 horas síncronas e 48 horas assíncronas, a formação busca não apenas aprimorar as habilidades técnicas dos educadores, mas também estimular uma reflexão crítica sobre novas metodologias que promovam o letramento estatístico e o pensamento computacional em sala de aula.

Ao final, os participantes receberam um certificado de curso de extensão universitária, validando não apenas sua formação técnica, mas seu engajamento em uma prática pedagógica reflexiva e inovadora.

1 CONTEXTUALIZAÇÃO DO PRODUTO EDUCACIONAL

A educação matemática no Brasil enfrenta desafios significativos, evidenciados pelo baixo desempenho de estudantes em avaliações nacionais e internacionais como o Programa Internacional de Avaliação de Estudantes (PISA)¹ (OCDE, 2022)² e o Sistema de Avaliação da Educação Básica (Saeb), particularmente na disciplina de Matemática. Existe uma lacuna nos investimentos financeiros e na aplicação equitativa de recursos na educação, além de uma defasagem nas práticas pedagógicas que muitas vezes não dialogam com as novas gerações e o crescente cenário digital. O déficit no aprendizado da matemática, especialmente em Probabilidade e Estatística, é uma preocupação, e a formação inicial de muitos docentes apresenta fragilidades nessa área, focando em conteúdos descritivos e carecendo de abordagens que integrem análises estatísticas avançadas e tecnologias digitais.

A Base Nacional Comum Curricular (BNCC)³, em suas diretrizes para o Ensino Médio, destaca a importância da "Matemática e suas Tecnologias", incentivando o uso de conceitos, procedimentos e estratégias para resolver problemas complexos e desenvolver o pensamento computacional. A competência EM13MAT405 (Utilizar conceitos iniciais de uma linguagem de programação na implementação de algoritmos escritos em linguagem corrente e/ou matemática), por exemplo, é central, pois propõe integrar tecnologias digitais, como a linguagem de programação, ao ensino da Matemática. No entanto, a realidade escolar frequentemente esbarra na ausência de infraestrutura tecnológica adequada e na falta de formação continuada que capacite os professores a utilizar essas ferramentas de forma crítica, significativa e ética.

É nesse contexto de desafios e necessidades que o presente produto educacional se insere. Ele surge como uma resposta direta à urgência de modernizar

¹ Oferece um estudo comparativo internacional sobre o desempenho dos estudantes na faixa etária dos 15 anos. Esse estudo é realizado a cada três anos pela Organização para a Cooperação e Desenvolvimento Econômico (OCDE). – BRASIL. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (Inep). Programa Internacional de Avaliação de Estudantes (Pisa). Brasília: MEC, 2024.

² OCDE: endereço eletrônico: <https://www.oecd.org/en.html>

³ Documento oficial que define os direitos de aprendizagem de todos os alunos das escolas brasileiras, elaborada pelo MEC. Disponível em: <http://portal.mec.gov.br/docman/dezembro-2018/pdf/104101-rcp004-18/file>. Acesso em: 6 out. 2025.

o ensino da Matemática, empoderando os docentes com uma metodologia ativa e relevante. Ao propor a programação em Python como alternativa didática, busca-se não apenas facilitar a compreensão de conceitos complexos de Probabilidade e Estatística, mas também despertar o interesse dos estudantes por carreiras em tecnologia e prepará-los para o exercício pleno da cidadania em um mundo cada vez mais digital.

2 O OBJETO DO PRODUTO EDUCACIONAL

O objeto central desse produto educacional é propor oficinas de capacitação em técnicas de programação aplicadas a Probabilidade e Estatística para aulas de Matemática no ensino médio.

Mais especificamente, o curso aborda os seguintes tópicos, integrando-os à programação em Python:

- **Fundamentos de Probabilidade:** O que é probabilidade, tipos de probabilidade, eventos na probabilidade, fórmula da probabilidade, usando espaços amostrais, operações básicas de conjuntos, aleatoriedade e simulação, regra de adição, regra de multiplicação (eventos dependentes e independentes), probabilidade condicional e independência.
- **Conceitos Básicos de Estatística:** Conceito básico de amostragem, tipos de variáveis (média, moda e mediana), medidas de dispersão.
- **Introdução e Técnicas de Programação em Python:** Introdução à linguagem Python e suas principais estruturas, técnicas de programação orientada a objetos aplicadas aos conceitos matemáticos.
- **Aplicação Prática:** Aulas expositivas dialogadas e a resolução de listas de exercícios contextualizados, que simulam situações reais e desafios do ensino de Probabilidade e Estatística no ensino médio, utilizando o Python como ferramenta de análise e visualização.

O objetivo é que os docentes, ao final da formação, não apenas dominem as técnicas de programação, mas também reflitam sobre como essas ferramentas podem enriquecer sua prática pedagógica, tornando o ensino da Matemática mais dinâmico, visual e compreensível para os estudantes, especialmente na abordagem de conceitos que historicamente apresentam complexidade e baixa proficiência.

3 METODOLOGIA APLICADA NO PRODUTO EDUCACIONAL

A metodologia do curso de extensão foi cuidadosamente delineada para promover um aprendizado ativo, colaborativo e reflexivo, ancorado na pesquisa narrativa que fundamentou a dissertação. Caracterizada por uma abordagem qualitativa de natureza aplicada e exploratória, a formação buscou gerar conhecimento e servir como insumo para novas práticas didáticas.

1. Formato Híbrido e Duração:

- **Carga Horária Total:** 60 horas.
- **Encontros Síncronos (12 horas):** Realizados em 3 sessões quinzenais (09/11, 23/11 e 07/12/2024) via plataforma online (Microsoft Teams⁴). Essas sessões permitiram a interação em tempo real, discussões e a resolução de dúvidas, alinhando-se à pesquisa narrativa que valoriza a escuta e a troca de experiências.
- **Atividades Assíncronas (48 horas):** Desenvolvidas em ambiente virtual de aprendizagem na modalidade à distância, proporcionando flexibilidade para os docentes aprofundarem os conteúdos e realizarem exercícios práticos.

2. Conteúdo Programático Modularizado:

- Os temas de Probabilidade e Estatística foram integrados à programação em Python, divididos em três eixos complementares para facilitar a aprendizagem progressiva. A apresentação do conteúdo foi feita via PowerPoint⁵, compartilhado no grupo de WhatsApp⁶, fomentando a construção colaborativa do conhecimento.

⁴ Plataforma unificada de comunicação e colaboração para videoconferências. **MICROSOFT. O que é o Microsoft Teams?** [S. I.], 2023. Disponível em: <https://support.microsoft.com/pt-br/topic/o-que-%C3%A9-o-microsoft-teams-3de4d369-0167-8def-b93b-0eb5286d7a29>. Acesso em 06 out. 2025.

⁵ MICROSOFT. O que é o PowerPoint? [S. I.], 2023. Disponível em: <https://support.microsoft.com/pt-br/office/o-que-%C3%A9-o-powerpoint-5f9cc860-d199-4d85-ad1b-4b74018acf5b>. Acesso em: 06 out. 2025.

⁶ WhatsApp: aplicativo de mensagens instantâneas. WHATSAPP. [S. I.], 2024. Disponível em: https://www.whatsapp.com/?lang=pt_br. Acesso em: 06 out. 2025.

3. Estratégias de Aprendizagem Ativa:

- **Listas de Exercícios Contextualizados:** Alinhados aos desafios do ensino de Probabilidade e Estatística no ensino médio, permitindo que os professores aplicassem os conhecimentos de programação em cenários práticos.
- **Plantões Diários de Dúvidas:** Oferecidos pela pesquisadora para criar um espaço de diálogo personalizado, essencial para superar inseguranças e promover a autonomia dos participantes.
- **Discussões e Trocas de Experiências:** No grupo de WhatsApp e durante os encontros síncronos, os docentes puderam compartilhar suas percepções, desafios e sucessos, criando uma "comunidade narrativa" que enriqueceu o processo formativo.

4. Recursos Tecnológicos e Acessibilidade:

- **Awesome Screenshot:** Visando garantir a ampla acessibilidade e a perenidade do conhecimento construído, todos os encontros síncronos da formação foram integralmente gravados. Este acervo digital transforma o curso em um recurso pedagógico de consulta contínua, permitindo que os participantes e outros interessados revisitem as discussões e as práticas de programação a qualquer momento. Todo o material audiovisual encontra-se permanentemente disponível, de forma gratuita, na plataforma Awesome Screenshot⁷ e pode ser acessado publicamente através do seguinte link: <https://www.awesomescreenshot.com/s/folder/F08xoly0E9/e494ab9bdd416f9bb4e27cb54ba76dc2>.

⁷ Plataforma assíncrona que visa potencializar a aprendizagem autônoma. AWESOME SCREENSHOT. Screenshot and screen video recorder. Disponível em: <https://www.awesomescreenshot.com>. Acesso em: 06 out. 2025.

- **Linguagem Python:** Escolhida por sua versatilidade, sintaxe prática e facilidade de compreensão, ideal para a introdução à programação em contextos educacionais.

5. Coleta de Narrativas e Análise Hermenêutica:

- Ao final de cada encontro, os participantes foram convidados a narrar suas expectativas, vivências e experiências por meio de questionários no Google Formulários⁸.
- A análise desses dados foi realizada sob uma abordagem hermenêutica, inspirada em Gadamer, Larrosa e Passeggi, buscando interpretar os sentidos atribuídos pelos professores ao uso da programação, as rupturas e resistências, e como a experiência se inscreveu em suas trajetórias docentes. Isso permitiu compreender que o medo inicial cedeu lugar à curiosidade e ao desejo de criar, transformando a relação dos docentes com a tecnologia e a didática da Matemática.

A eficácia desta abordagem ficou evidente nas narrativas dos próprios participantes. O percurso formativo transformou a insegurança inicial, capturada em relatos como “Nunca pensei que fosse conseguir escrever um código”, em um sentimento de autoria pedagógica e reinvenção. Essa jornada, do receio à apropriação da tecnologia, comprova o potencial da formação para além da técnica, focando na ressignificação da prática docente.

A metodologia, portanto, não se limitou à transmissão de conhecimentos técnicos, mas buscou uma transformação pedagógica profunda, onde o professor é visto como um sujeito reflexivo e protagonista de sua própria formação.

⁸ Aplicativo de gerenciamento de pesquisas lançado pelo Google. - Conteúdo aberto. In: WIKIPÉDIA: a enciclopédia livre. Disponível em: https://pt.wikipedia.org/wiki/Google_Forms. Acesso em: 06 out. 2025

4 SOBRE AS AUTORAS

ALESSANDRA SOARES DOS SANTOS

- Mestre em Educação na Universidade Municipal de São Caetano do Sul 2023 - 2025 – Brasil.
- Master em Information Systems em Arquitetura e Engenharia de Software na Universidade Municipal de São Caetano do Sul 2021 a 2022 - Brasil
- Pós-Graduação em Segurança da Informação na Faculdade Impacta 2012 a 2013 Brasil.
- Graduação em Análise e Desenvolvimento de Sistemas na Faculdade Claretiano 2009 a 2011.
- Possuo experiência sólida como docente, mentora educacional e produtora de conteúdo, lecionando uma vasta gama de disciplinas que incluem desenvolvimento front-end e back-end, banco de dados, computação em nuvem, lógica de programação, padrões de projeto e arquitetura, e metodologias ágeis.
- Arquiteta de Software com 15 anos de experiência em empresas de grande porte, especializada em engenharia de sistemas complexos, arquitetura corporativa e soluções cloud native.

PROFA. DRA. ADRIANA BARROSO DE AZEVEDO

- Possui graduação em Pedagogia pela Universidade Federal de Mato Grosso (1993).
- Mestrado em Educação pela Universidade Federal de Mato Grosso (1997).
- Doutorado em Comunicação Social pela Universidade Metodista de São Paulo (2002).
- Pós-doutorado em Educação pela Universidade Federal do Rio Grande do Norte (2015).
- Professora titular no PPGE da Universidade Metodista de São Paulo.
- Pró-reitora de pós-graduação e pesquisa da UMESP.
- Diretora Nacional de Educação da Rede Metodista de Educação.
- Experiência na área de Educação, educação a distância, gestão do ensino superior.
- Pesquisadora narrativa atuando principalmente nos seguintes temas: educação a distância, educação básica, educação inclusiva, tecnologias digitais de informação e comunicação e epistemologia da pesquisa narrativa.

5 CONSIDERAÇÕES FINAIS

O produto educacional "LINGUAGEM PYTHON NA SALA DE AULA: PRÁTICAS INOVADORAS NO ENSINO DE PROBABILIDADE E ESTATÍSTICA NO ENSINO MÉDIO", demonstrou ser uma contribuição significativa e tangível para a formação docente e para a educação matemática brasileira. A pesquisa subjacente, pautada em uma abordagem narrativa e hermenêutica, revelou que, embora o medo inicial e as barreiras estruturais sejam reais, a experiência de formação é capaz de despertar nos professores um novo olhar para a tecnologia e para o ensino da Matemática.

A integração da programação em Python no currículo da Matemática, conforme proposto e executado neste curso, provou ser uma mediação potente entre os saberes escolares e as demandas do mundo contemporâneo. Os docentes não apenas adquiriram habilidades técnicas, mas também desenvolveram uma postura investigativa, compreendendo que a tecnologia, quando mediada por uma intencionalidade pedagógica clara, pode ressignificar o ensino, tornando conceitos abstratos mais visuais, contextualizados e interativos.

Embora este produto educacional tenha sido um sucesso em despertar a "esperança pedagógica" e promover "pequenas revoluções pedagógicas", reforçam que sua consolidação em larga escala exige investimentos contínuos em infraestrutura tecnológica e políticas públicas que apoiem a formação continuada de professores. A valorização do diálogo entre pares, a experimentação metodológica e a escuta das experiências vividas são cruciais para que a inovação não resida apenas na grandiosidade do recurso tecnológico, mas na forma como ele é apropriado para transformar as práticas em sala de aula.

Mais do que uma capacitação técnica, o curso se revelou uma experiência formadora. Observou-se um claro movimento de transformação nos participantes: o medo inicial da programação deu lugar à curiosidade; a desconfiança cedeu espaço ao desejo de criar; e a sensação de inadequação foi substituída por um crescente senso de autoria pedagógica. Este produto educacional, portanto, não entrega apenas um método, mas a evidência de que é possível, através da formação continuada e

dialógica, transformar a relação dos professores com a tecnologia e, conseqüentemente, o futuro do ensino da Matemática.

Portanto, não é um ponto final, mas um novo começo. Ele estabelece uma base sólida para futuras expansões, adaptabilidade a outros conteúdos da Matemática e a construção de comunidades de prática que continuarão a enriquecer a jornada de educadores e estudantes rumo a um ensino mais engajador e eficaz.

REFERÊNCIAS

- ALMEIDA, Cínthia Soares de. **Dificuldades de aprendizagem em matemática e a percepção dos professores em relação a fatores associados ao insucesso nesta área.** 2006. Monografia (Graduação em Educação) - Universidade Católica de Brasília, Brasília, 2006.
- ALMEIDA, Maria Elizabeth Bianconcini; VALENTE, José Armando. **Tecnologias e currículo: Trajetórias convergentes ou divergentes?** São Paulo: Paulus, 2011.
- ALMEIDA, Tamiris. **Pisa 2022: Por que o Brasil está nas últimas posições em matemática, ciências e leitura?** 2023. Disponível em: <https://futura.frm.org.br/conteudo/educacao-basica/noticia/pisa-2022-por-que-o-brasil-esta-nas-ultimas-posicoes-em-matematica-leitura-ciencias>. Acesso em: 06 out. 2025
- AZEVEDO, Adriana Barroso de; PASSEGGI, Maria da Conceição. **Narrativas das Experiências Docentes com o uso de Tecnologias na Educação.** São Bernardo do Campo: Universidade Metodista de São Paulo, 2016
- BAYER, Arno *et al.* **Probabilidade na escola.** In: III CONGRESSO INTERNACIONAL DE ENSINO DE MATEMÁTICA, v. 1., 2005. p. 1-12.
- BERNARDES, O . Para uma abordagem do conceito de probabilidade. **Educação e Matemática.** nº 3, 1987.
- BITTAR, Marilena; GUIMARÃES, Sheila Denize; VASCONCELLOS, Monica. A integração da tecnologia na prática do professor que ensina matemática na educação básica: uma proposta de pesquisa-ação. **REVEMAT: Revista Eletrônica de matemática**, Florianópolis, v. 3, n. 1, 2008 p. 84-94, 26 de mar de 2008
- BARICHELO, Leonardo. **Pensamento Computacional.** São Paulo: Instituto Nacional de Matemática Pura e Aplicada, 2021. 67 p.
- BRASIL. Ministério da Educação. **Base Nacional Comum Curricular.** Brasília, DF: MEC, 2018.: MEC, 2018
- BRASIL. Presidência da República. Casa Civil. **O Brasil e a OCDE.** Disponível em: <https://www.gov.br/casacivil/pt-br/assuntos/colegiados/ocde/apresentacao?form=MG0AV3>. Acesso em: 06 out. 2025.
- BRASIL. Ministério da Educação. **Pisa 2018 revela baixo desempenho escolar em Leitura, Matemática e Ciências no Brasil.** Portal do Ministério da Educação. Brasília, DF: MEC, 2019. Disponível em: <http://portal.mec.gov.br/ultimas-noticias/211-218175739/83191-pisa-2018-revela-baixo-desempenho-escolar-em-leitura-matematica-e-ciencias-no-brasil>. Acesso em: 06 out. 2025.

BRASIL. Ministério da Educação. Conselho Nacional de Educação. **Resolução nº 4, de 17 de dezembro de 2018**. Institui a Base Nacional Comum Curricular na Etapa do

BRASIL. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira – Inep. Diretoria de Avaliação da Educação Básica. **Sistema de Avaliação da Educação Básica: documentos de referência**. Versão 1.0. Brasília, DF: MEC/INEP/DAEB, 2018a.

CLANDININ, D. J.; CONNELLY, F. M. **Pesquisa narrativa: experiências e história na pesquisa qualitativa**. Uberlândia: EDUFU, 2011.

COSTA, N. M. L. DA; PRADO, M. E. B. B. A Integração das Tecnologias Digitais ao Ensino de Matemática: desafio constante no cotidiano escolar do professor. **Perspectivas da Educação Matemática**, Campo Grande, v. 8, n. 16, 06 de nov de 2015

CRUZ, Elaine Patricia. SP: ensino médio têm pior desempenho em matemática em 11 anos. **Agência Brasil**. São Paulo, 02 de mar de 2022. Disponível em: <https://agenciabrasil.ebc.com.br/educacao/noticia/2022-03/sp-ensino-medio-tem-o-pior-desempenho-em-matematica-em-11-anos>. Acesso em: 06 out. 2025

Ensino Médio (BNCC-EM), como etapa final da Educação Básica, nos termos do artigo 35 da LDB, completando o conjunto constituído pela BNCC da Educação Infantil e do Ensino Fundamental, com base na Resolução CNE/CP nº 2/2017, fundamentada no Parecer CNE/CP nº 15/2017. **Diário Oficial da União**, Brasília, DF: MEC, 18 dez. 2018. Disponível em: <http://portal.mec.gov.br/docman/dezembro-2018-pdf/104101-rcp004-18/file>. Acesso em: 06 out. 2025.

GADAMER, Hans-Georg. **Verdade e Método: Traços fundamentais de uma hermenêutica filosófica**. Petrópolis: Vozes, 1999.

GALILEU SISTEMA DE GESTÃO ESCOLAR. **A Importância do Lúdico no Ensino de Matemática**. Disponível em: <https://www.sistemagalileu.com.br/a-importancia-do-ludico-no-ensino-de-matematica/>. Acesso em: 06 out. 2025.

HABOWSKI, Adilson Cristiano; CONTE, Elaine. **As Tecnologias na Educação: (re)pensando seus sentidos tecnopoéticos**. São Paulo: Pimenta Cultural, 2019

LARROSA, J. Notas sobre a experiência e o saber da experiência. **Revista Brasileira de Educação**. Campinas, n. 19, p. 20-28, jan./fev./mar./abr. 2002.

LARROSA, Jorge. **Tremores: escritos sobre experiência**. Tradução: Cristina Antunes e João Wanderley Geraldi. 1. ed. Belo Horizonte: Autêntica, 2014.

LOPES, Celi Aparecida Espasandin. A Estatística e a probabilidade no currículo da escola básica e a formação dos professores. **IX Seminário IASI de Estatística Aplicada**. Rio de Janeiro, 2003.

LOPES, C. E. (2008). O ensino da estatística e da probabilidade na educação básica e a formação dos professores. **Cadernos CEDES**, vol. 28, n. 74, p. 57-73, 2008.

LOPES, Celi Espasandin. **O ensino da estatística e da probabilidade na Educação Básica: uma análise a partir da prática docente**. Porto Alegre: Ed. Penso, 2010.

MARINHO, Simão Pedro; LOBATO, Wolney. Tecnologias digitais na educação: desafios para a pesquisa na pós-graduação em educação. **Colóquio de Pesquisa em Educação**, v. 6, p. 1-9, 2008

MORAN, José. **Metodologias Ativas para uma Educação Inovadora**. Penso Editora, 2018.

PAPERT, Seymour. Logo: computadores e educação. Tradução de José A. Valente. São Paulo: Brasiliense, 1985.

PAPERT, Seymour. Mindstorms: Children, Computers, and Powerful Ideas. Basic Books, 1980.

PASSEGGI, M.C. Injunção institucional e sedução autobiográfica: as faces autopoiética e avaliativa dos memoriais. In: BARBOSA, T. M. N., PASSEGGI, M. C. **Memorial Acadêmico - gênero, injunção institucional, sedução autobiográfica**. Natal: EDUFRN, 2011. p. 19-39.

PESENTE, Guilherme Moraes. **O ensino de matemática por meio de linguagem de programação Python**. 2019. Dissertação (Mestrado em Ensino de Ciência e Tecnologia) – Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2019. Disponível em: <https://repositorio.utfpr.edu.br/jspui/handle/1/5020>. Acesso em: 06 out. 2025.

Relatório Institucional – Ano 2021. **INESC - Instituto de Estudos Socioeconômicos**. Disponível em: https://inesc.org.br/wp-content/uploads/2022/05/Relatorio-institucional-2021_V2.pdf?x59185. Acesso em: 06 out. 2025.

SILVA, Nelson Antônio da; FIGUEIREDO, Helenara R. Sampaio. A Educação Estatística Na Educação Básica Do Brasil, Estados Unidos, França E Espanha Segundo Os Documentos Curriculares. **Revista Eletrônica de Educação Matemática**, v. 14 2019. Disponível em: <https://periodicos.ufsc.br/index.php/revemat/article/view/1981-1322.2019.e62813>. Acesso em: 06 out. 2025.

SILVEIRA, B. **As Dificuldades de Aprendizagem na Matemática: Discursos Legitimados por estudantes de Educação Básica**. Caçapava do Sul: Universidade Federal do Pampa, 2014.

SOUZA NETO, Alaim. **Do aprender ao ensinar com as tecnologias digitais: discussões atuais aos professores**. São Paulo: Pimenta Cultural, 2016.

TARDIF, Maurice. **Saberes docentes e formação profissional**. 17. ed. Petrópolis: Vozes, 2014.

VALENTE, J. A. **Pensamento computacional na educação**. São Paulo: Penso, 2021

APÊNDICE A – FORMULÁRIO DE INSCRIÇÃO

TÉCNICAS DE PROGRAMAÇÃO SOBRE PROBABILIDADE E ESTATÍSTICA PARA AULA DE MATEMÁTICA DO ENSINO MÉDIO FORMULÁRIO DE INSCRIÇÃO

Evento: on-line

Data: 09/11, 23/11 e 07/12/2024

Horário: 10h às 12h (horário de Brasília)

Total de vagas: 80

Investimento: gratuito

OBS.: O link para participar dos 3 encontros on-line serão enviados pelo Whatsapp.

E-mail: _____

Nome: _____

Número do celular?: _____

Quais são as escolas e anos letivos que você leciona a disciplina de matemática:

Você participará dos 3 encontros on-line?

dias 09/11, 23/11 e 07/12/2024 (19h às 22h): () Sim () Não

Relate quais as dificuldades que os estudantes têm apresentado durante o ensino dos conteúdos de probabilidade e estatística em sala de aula?

Quais os motivos pelo qual você decidiu realizar essa formação?

APÊNDICE B – FORMULÁRIO DE FEEDBACK DO 1º. ENCONTRO

Queremos saber como foi a sua experiência na formação: **TÉCNICAS DE PROGRAMAÇÃO SOBRE PROBABILIDADE E ESTATÍSTICA PARA AULA DE MATEMÁTICA DO ENSINO MÉDIO**, realizada no dia 09/11/2024.

Responda a esta pesquisa rápida e conte-nos sua opinião. As respostas serão anônimas.

E-mail: _____

Nome: _____

Qual é a sua cidade e Estado? _____

Qual a sua idade? _____

Qual é o seu gênero? () Feminino () Masculino () Prefiro não informar

Quais as suas expectativas sobre as possibilidades de ensinar e aprender

Matemática a partir do uso de programação?

APÊNDICE C – FORMULÁRIO DE FEEDBACK DO 2º. ENCONTRO

Descreva como foi a sua experiência na formação: **TÉCNICAS DE PROGRAMAÇÃO SOBRE PROBABILIDADE E ESTATÍSTICA PARA AULA DE MATEMÁTICA DO ENSINO MÉDIO**, realizada no dia 23/11/2024 - das 10h até 12h.

As respostas serão anônimas.

E-mail: _____

Nome: _____

Qual é a sua cidade e Estado? _____

Qual a sua idade? _____

Qual é o seu gênero? () Feminino () Masculino () Prefiro não informar

Quais as suas expectativas sobre as possibilidades de ensinar e aprender Matemática a partir do uso de programação?

APÊNDICE D – FORMULÁRIO DE FEEDBACK DO EVENTO

Queremos saber como foi a sua experiência na formação: **TÉCNICAS DE PROGRAMAÇÃO SOBRE PROBABILIDADE E ESTATÍSTICAS PARA AULA DE MATEMÁTICA DO ENSINO MÉDIO.**

Responda a esta pesquisa rápida e conte-nos sua opinião. As respostas serão anônimas.

E-mail: _____

Nome: _____

Qual é a sua cidade e Estado? _____

Telefone para contato: _____

Comente como os conteúdos apresentados irão ajudar você no ensino da disciplina de matemática? _____

Você ficou satisfeito com:

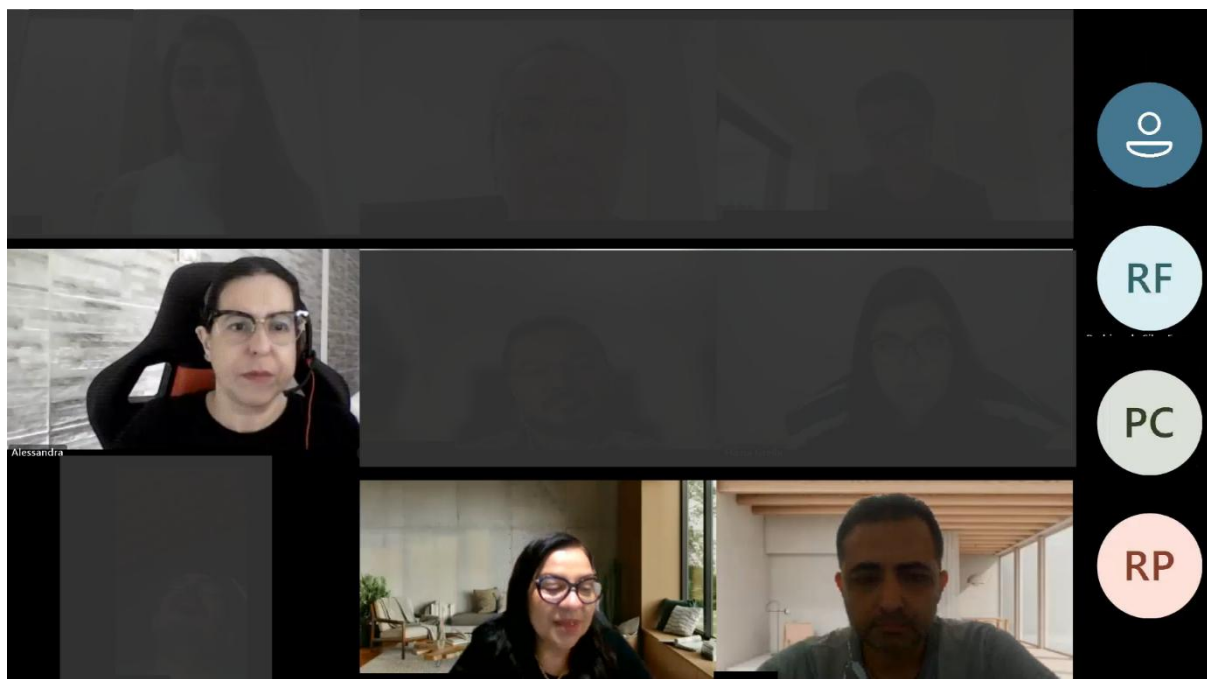
(Onde 1 = Muito insatisfeito e 5 = Muito satisfeito)

- | | |
|--|-------------------------------|
| - conteúdo didático: | () 1 () 2 () 3 () 4 () 5 |
| - exemplos abordados no curso: | () 1 () 2 () 3 () 4 () 5 |
| - didática do palestrante: | () 1 () 2 () 3 () 4 () 5 |
| - carga horária: | () 1 () 2 () 3 () 4 () 5 |
| - atividades complementares: | () 1 () 2 () 3 () 4 () 5 |
| - satisfação geral com a formação de hoje: | () 1 () 2 () 3 () 4 () 5 |

Narre como foi a sua experiência durante essa formação e como ela poderá contribuir para o ensino desses conteúdos didáticos durante as aulas?

APÊNDICE E – MOMENTOS DA FORMAÇÃO





7 Técnicas de programação orientadas à objetos

➤ **Orientação à objetos:** organiza o código em torno de objetos, que são representações de entidades do mundo real. Os conceitos básicos da orientação a objetos são:

➤ **Objetos:** São instâncias de uma classe, e são compostos por atributos e métodos.

➤ **Atributos:** são características do objetos

➤ **Métodos:** são as ações que o objeto pode realizar.

➤ **Classes:** São os modelos que dipificam o objeto.

Na classe teremos atributos e métodos do objeto

➤ **Atributos:** São as variáveis de classe, ou seja, eles definem a estrutura de uma classe.

As variáveis são pequenos espaços de memória reservados para armazenar dados.

```
import matplotlib.pyplot as plt
# Dados para o eixo x (anos)
anos = [2010, 2011, 2012, 2013, 2014, 2015]

# Dados para o eixo y (receitas)
receitas = [5000, 8000, 6500, 9000, 7500, 10000]

# Criando o gráfico de linhas
plt.plot(anos, receitas)

# Configurando rótulos e título do gráfico
plt.xlabel('Ano')
plt.ylabel('Receitas (em dólares)')
plt.title('Receitas ao longo dos anos')

# Exibindo o gráfico
plt.show()
```



The screenshot shows a video call interface with two participants, Alessandra, visible in a vertical strip on the right. The main area displays two web browsers side-by-side.

The top browser is **OnlineGDB** (onlinegdb.com). It shows a Python 3 code editor with the following code:

```
1 ...
2
3 Welcome to GDB Online.
4 GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,
5 C#, OCaml, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog
6 Code, Compile, Run and Debug online from anywhere in world.
7
8 ...
9 print ('Hello World!')
```

The bottom browser is **myCompiler** (mycompiler.io/pt/new/python). It shows a Python script that generates a bar chart:

```
1 import matplotlib.pyplot as plt
2
3 categorias = ['Matemática', 'Português']
4 valores = [5, 4]
5
6 plt.bar(categorias, valores)
7
8 plt.xlabel('Categorias')
9 plt.ylabel('valores')
10 plt.title('Disciplinas')
11
12 plt.show()
```

The output of the program is a bar chart titled "Disciplinas" showing two bars: "Matemática" with a value of 5 and "Português" with a value of 4.

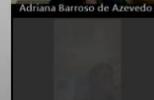
Disciplina	Valor
Matemática	5
Português	4

A woman with dark hair and glasses is speaking in a video call. She is wearing a dark top. The background is a blurred indoor setting with a sofa and a window.

A +2



Alessandra

A small video window showing a woman with dark hair and glasses, identified as Alessandra, in a home setting.

HelloWorld.py

NEW

PYTHON ▾

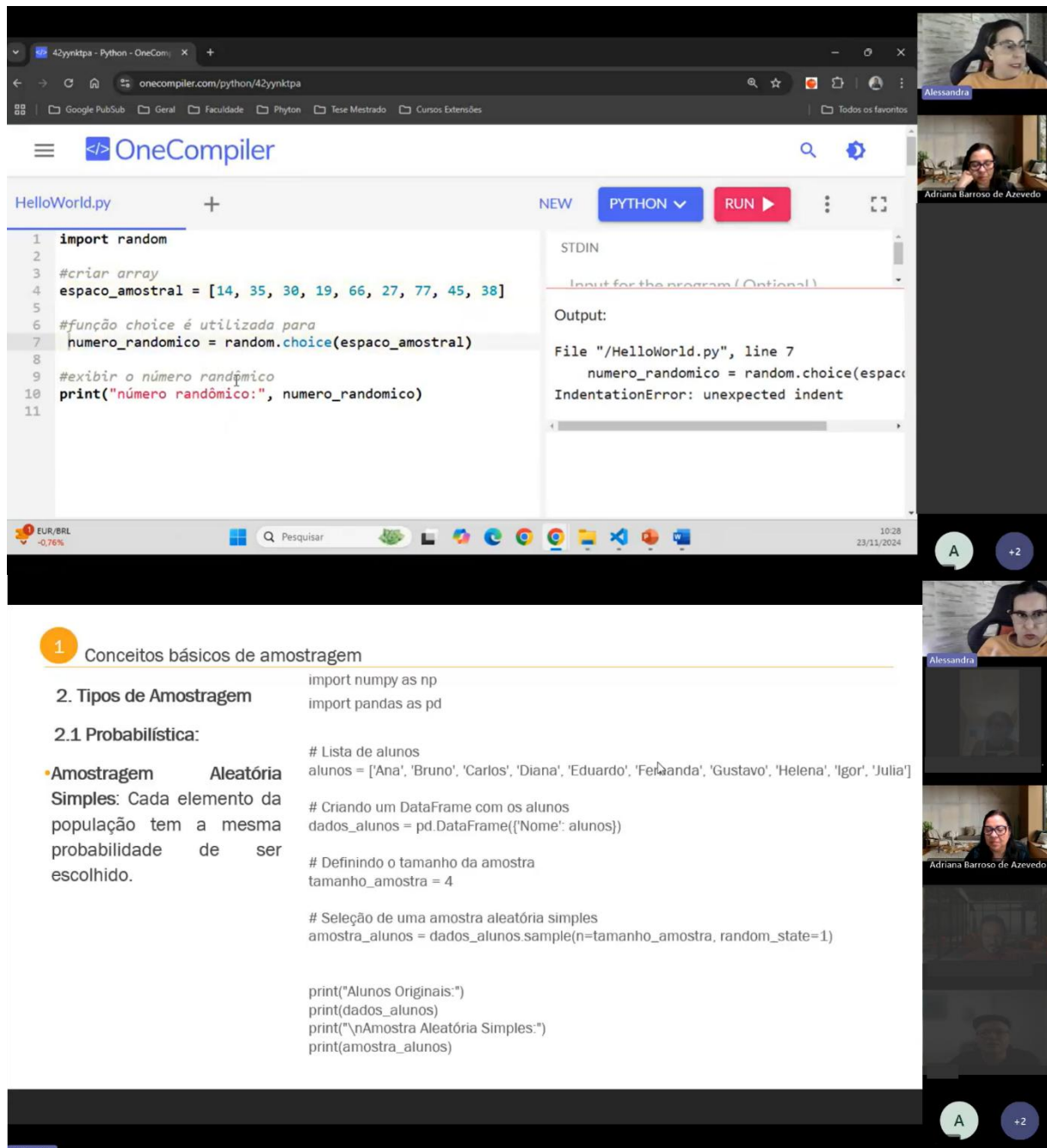
RUN ▶

```
1 import random
2
3 #criar array
4 espaco_amostral = [14, 35, 30, 19, 66, 27, 77, 45, 38]
5
6 #função choice é utilizada para
7 numero_randomico = random.choice(espaco_amostral)
8
9 #exibir o número randômico
10 print("número randômico:", numero_randomico)
11
```

STDIN

Output:

número randômico: 27



The image shows a OneCompiler Python IDE interface. The main editor displays a file named `HelloWorld.py` with the following Python code:

```

1 import random
2
3 #criar array
4 espaco_amostral = [14, 35, 30, 19, 66, 27, 77, 45, 38]
5
6 #função choice é utilizada para
7 numero_randomico = random.choice(espaco_amostral)
8
9 #exibir o número randômico
10 print("número randômico:", numero_randomico)
11

```

The output pane on the right shows the error message: `IndentationError: unexpected indent` at line 7. The error message is partially obscured by the text `numero_randomico = random.choice(espaco_amostral)`.

Below the IDE, a presentation slide titled "1 Conceitos básicos de amostragem" is visible. The slide content is as follows:

2. Tipos de Amostragem

2.1 Probabilística:

- **Amostragem Aleatória Simples:** Cada elemento da população tem a mesma probabilidade de ser escolhido.

```

import numpy as np
import pandas as pd

# Lista de alunos
alunos = ['Ana', 'Bruno', 'Carlos', 'Diana', 'Eduardo', 'Fernanda', 'Gustavo', 'Helena', 'Igor', 'Julia']

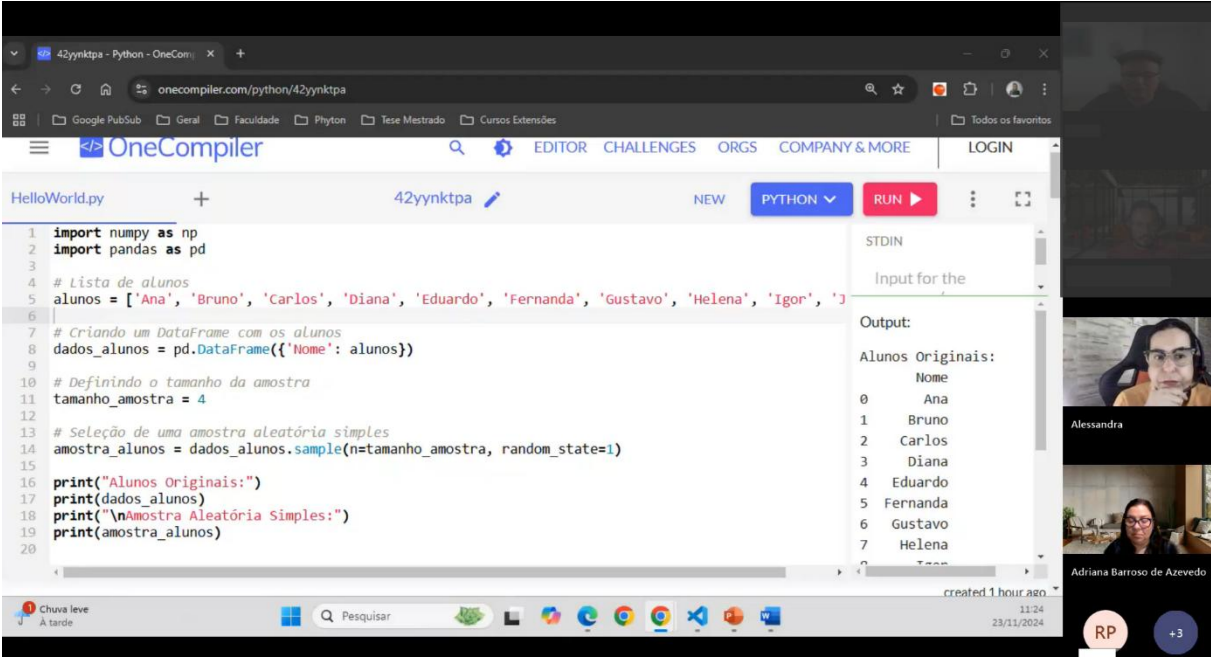
# Criando um DataFrame com os alunos
dados_alunos = pd.DataFrame({'Nome': alunos})

# Definindo o tamanho da amostra
tamanho_amostra = 4

# Seleção de uma amostra aleatória simples
amostra_alunos = dados_alunos.sample(n=tamanho_amostra, random_state=1)

print("Alunos Originais:")
print(dados_alunos)
print("\nAmostra Aleatória Simples:")
print(amostra_alunos)

```



The screenshot displays a web-based Python IDE interface. The main editor shows a Python script that uses `numpy` and `pandas` to create a DataFrame of student names, sample 4 students, and print the results. The output panel on the right shows the execution results, including the original list of students and the sampled students.

```
1 import numpy as np
2 import pandas as pd
3
4 # Lista de alunos
5 alunos = ['Ana', 'Bruno', 'Carlos', 'Diana', 'Eduardo', 'Fernanda', 'Gustavo', 'Helena', 'Igor', 'João']
6
7 # Criando um DataFrame com os alunos
8 dados_alunos = pd.DataFrame({'Nome': alunos})
9
10 # Definindo o tamanho da amostra
11 tamanho_amostra = 4
12
13 # Seleção de uma amostra aleatória simples
14 amostra_alunos = dados_alunos.sample(n=tamanho_amostra, random_state=1)
15
16 print("Alunos Originais:")
17 print(dados_alunos)
18 print("\nAmostra Aleatória Simples:")
19 print(amostra_alunos)
20
```

Output:

Alunos Originais:

	Nome
0	Ana
1	Bruno
2	Carlos
3	Diana
4	Eduardo
5	Fernanda
6	Gustavo
7	Helena
8	Igor
9	João

Amostra Aleatória Simples:

	Nome
0	Ana
1	Bruno
2	Carlos
3	Diana

created 1 hour ago

Chuva leve
À tarde

Pesquisar

11:24
23/11/2024

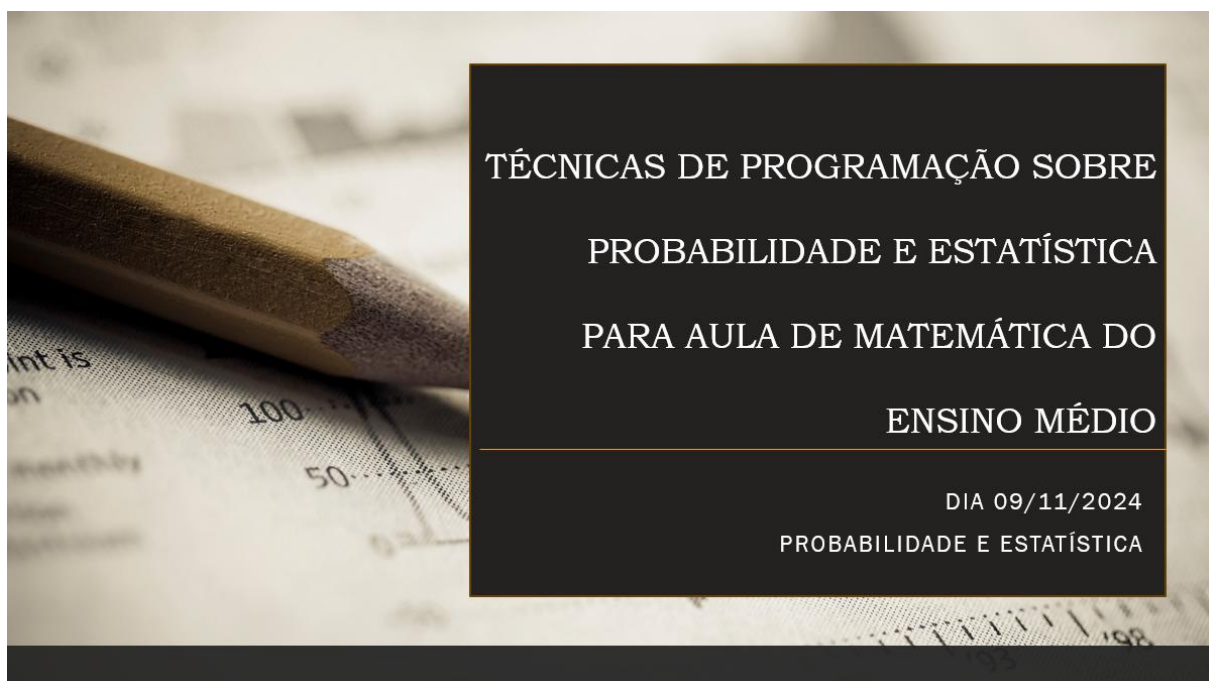
RP +3

Alessandra

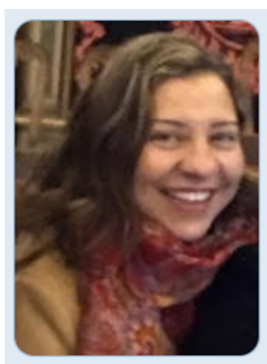
Adriana Barroso de Azevedo

APÊNDICE F – CONTEÚDO PROGRAMÁTICO DO 1º ENCONTRO

Este apêndice tem o objetivo de apresentar o conteúdo programático compartilhado com os participantes do primeiro encontro do evento de extensão intitulado: “técnicas de programação sobre Probabilidade e Estatística para aula de Matemática do ensino médio”.



Apresentação - Orientadora



Dra. Adriana Barroso de Azevedo

Possui graduação em Pedagogia pela Universidade Federal de Mato Grosso (1993), mestrado em Educação pela Universidade Federal de Mato Grosso (1997) e doutorado em Comunicação Social pela Universidade Metodista de São Paulo (2002), pós doutorado em Educação pela Universidade Federal do Rio Grande do Norte (2015). Atualmente é professora titular no PPGE da Universidade Metodista de São Paulo e professora convidada no PPGE da Universidade Municipal de São Caetano do Sul. É diretora Nacional de Educação da Rede Metodista de Educação. Tem experiência na área de Educação, é pesquisadora Narrativa atuando principalmente nos seguintes temas: educação a distância, educação básica, educação inclusiva, tecnologias digitais de informação e comunicação e epistemologia da pesquisa narrativa.

Apresentação

César Augusto do Prado Moraes



Possui graduação em Matemática pela Fundação Educacional de Penápolis (2004); Pedagogia pela Universidade Nove de Julho (2012); Letras - Português/Inglês(2019), História(2021) e Ciências Biológicas(2021) pelo Centro Universitário de Jales e Geografia pelo Centro Universitário FAVENI(2022); especializações em Metodologia do Ensino e Aprendizagem da Matemática pela Faculdade São Luís (2007) , Educação e Neurociências pela Faculdades Integradas Campos Salles (2019), Educação Inclusiva com Ênfase em Autismo pela Faculdade Casa Branca (2021) e Educação Matemática pela Faculdade Unida de São Paulo (2022), mestrado (2010) e doutorado (2018) em Educação pela Universidade Metodista de São Paulo e Pós-doutorado em Docência em Educação em Ciências e Matemática pela Universidade Federal do Pará (2021). Atualmente é professor universitário adjunto na Universidade Federal do Piauí, campus Bom Jesus- PI. Tem experiência na área de Matemática e Pedagogia, atuando principalmente nos seguintes temas: Estatística, Bioestatística, Matemática Financeira, Avaliação Educacional , Gestão Escolar , Metodologia do Ensino de Ciências e Matemática ,Tecnologia da Educação, História da Educação, Filosofia da Educação, Psicologia da Educação e Ética na Educação. Professor Colaborador do Programa de Pós-Graduação em Educação da Universidade Metodista de São Paulo na linha de Formação de Educadores.

Apresentação – Mestrando

Alessandra Soares dos Santos



Possui graduação em Análise e Desenvolvimento de Sistemas pela Faculdade Claretiano (2011), pós-graduação em Segurança da Informação pela Faculdade Impacta (2013), pós-graduação em Master em Information Systems em Arquitetura e Engenharia de Software pela Universidade Municipal de São Caetano do Sul (2023), mestrando em Educação pela Universidade Municipal de São Caetano do Sul. É produtora de material didático nas disciplinas de Tecnologia da Informação. Possui experiência em docência nas disciplinas de computação pelo Senac Vila Prudente, Codenation e Senai Santa Catarina. É Arquiteta de Tecnologia no Banco BV.

OBJETIVO DESTA FORMAÇÃO

Apresentar novas ferramentas didáticas para o ensino da Matemática e, principalmente, nos conteúdos de Estatística e Probabilidade

Aprender
linguagem
Phyton

Utilizar Phyton
para
resoluções de
Matemática

Utilizar essa
ferramenta
didática em
sala de aula

AGENDA

- 1 O que é probabilidade?
 - ponto amostral
 - espaço amostral
- 2 Tipos de probabilidade
- 3 Eventos na probabilidade
- 4 Fórmula da probabilidade
- 5 Como calcular a probabilidade
- 6 Diferenças entre probabilidade e estatística
- 7 Técnicas de programação orientada à objetos
- 8 Introdução à linguagem Phyton e suas principais estruturas
- 9 Exercícios

1 O que é probabilidade?

- A a frequência relativa que se espera que um evento ocorra.
- De acordo com (LOPES, MEIRELLES; 2005, p.1), suas raízes apareceram principalmente nos jogos e apostas.
Outra forma de explicar a probabilidade,

"A probabilidade além de ter suas raízes na solução de problemas de jogos também as tem no processamento de dados estatísticos. Os problemas estatísticos mais importantes que requerem o pensamento probabilístico originam-se no processo de amostras." (LOPES, MEIRELLES; 2005, p.1)

➔ Exemplo:

Em um saco com 692 M&Ms você encontrou 151 azuis.

Qual a porcentagem de M&M azuis?

Se retirar um M&M qual a chance de que ele seja azul?

1 O que é probabilidade?

➔ Proposta de resolução:

Em um saco com 692 M&Ms você encontrou 151 azuis.

Qual a porcentagem de M&M azuis?

$$151/692=0,218= 22\%$$

Se retirar um M&M qual a chance de que ele seja azul?

22%

Como utilizar técnicas de programação para explicar esse conceito / resolução?



1 O que é probabilidade?

- **Ponto amostral:** é uma das possíveis ocorrências ou resultados do experimento.
- **Experimento:** são processos que, mesmo repetidos várias vezes sob condições semelhantes, apresentam resultados imprevisíveis.
- **Espaço amostral (Ω ou S):** é um conjunto formado por todos os pontos amostrais de um experimento. É o conjunto de todos os possíveis resultados de um experimento.

➡ **Exemplo:**

Determine o espaço amostral do seguinte experimento:

a) lançamento de uma moeda

1 O que é probabilidade?

➡ **Proposta de resolução:**

Determine o espaço amostral do seguinte experimento:

a) lançamento de uma moeda ➡ **Experimento**
 resolução: {Ca, Co}; ➡ **Espaço Amostral**
 Ponto amostral

VAMOS APRENDER
TÉCNICAS DE
PROGRAMAÇÃO PARA
DESENVOLVER ESSE
CONCEITO

2 Tipos de probabilidade

- **Clássica:** utilizada quando todos os resultados possíveis são igualmente prováveis e há um número finito de casos possíveis (resultado igualmente prováveis).

Fórmula: $P = \text{número casos favoráveis} / \text{número de casos possíveis}$

Proposta de Resolução: Um Dado possui o espaço amostral {1,2,3,4,5,6}.

Se eu escolher o número 3, qual é o resultado:

$$P(3) = 1/6 \rightarrow 0,167$$

- **Empírica:** conhecida também como frequentista, é calculada a partir de repetições do experimento e análise dos resultados (baseada em uma pesquisa).

Fórmula: $P = \text{número de ocorrências do evento} / \text{total de observações}$

Proposta de Resolução: Um dado é lançado 100 vezes. Se o número 4 aparecer 25 vezes, a probabilidade empírica de obter um 4 em um lançamento seria:

$$P(4) = 25/100 = 0,25 \text{ ou } 25\%$$

2 Tipos de probabilidade

- **Subjetiva:** resulta da intuição, de suposições fundamentadas e estimativas (palpite).

Fórmula: $P = \text{proposta do evento ocorrer} / \text{total de observações}$

Proposta de Resolução: Em uma pesquisa Gallup (empresa americana de pesquisa de opiniões), entre 855 adultos escolhidos aleatoriamente, 710 confirmaram ter voado em algum avião comercial. Como representar essa probabilidade:

$$P(\text{voar em avião comercial}) = 710 / 855 \rightarrow 0,83\%$$

3 Eventos na probabilidade

➤ **Simples:** possui apenas um ponto amostral.

Exemplo: Sair “cara” em um lançamento de uma moeda

O evento de sair cara possui um único elemento. $E = \{\text{cara}\}$

O seu número de elementos é $n(E) = 1$

➤ **Certo:** evento que é igual ao espaço amostral, com probabilidade de ocorrência de 100%.

Exemplo: uma delegação feminina de futebol tem 32 jogadoras. Qual a probabilidade de uma delas ser sorteada e ser mulher:

$$P = 32 / 32 = 1 \rightarrow 100\%$$

3 Eventos na probabilidade

➤ **Impossível:** evento que é igual ao conjunto vazio, com 0% de chances de ocorrência

Exemplo: Imagine que temos uma caixa com bolas numeradas de 1 a 20 e que todas as bolas são vermelhas. O evento “tirar um número maior que 21” é impossível, visto que o maior número é 20.

$$P = 21 / 0 = 0$$

➤ **Provável:** evento que tem possibilidade de acontecer, com chances de 0 a 1, ou seja, de 0% a 100%.

Exemplo: Em um lançamento de dados, qual a chance de sair um número par?

Proposta de Resolução: Um Dado possui o espaço amostral {1,2,3,4,5,6}.

$$P(\text{par}) = 3/6 \rightarrow 0,5$$

4 Fórmula da probabilidade

$$P(A) = n(A) / n(S), \text{ sendo:}$$

P(A): probabilidade do evento A ocorrer

n(S ou Ω): número total de casos possíveis (número de elementos do espaço amostral)

n(A): número de casos favoráveis ao evento

5 Como calcular a probabilidade

➤ Cálculo da probabilidade: é realizado pela razão entre o número de casos favoráveis ao evento e o número de casos possíveis:

Exemplo: um baralho comum apresenta 52 cartas, 4 de cada naipe: ouros, copas, espadas e paus. Ao se tirar 1 carta de um baralho, qual é a probabilidade de ela ser um rei? Temos 4 reis, sendo esse o número de elementos de nosso evento.

$$P(\text{rei}) = 4 / 52 = 7,69\%$$

Exemplo: Ao se retirar uae um baralho, qual é a probabilidade de sair uma carta de ouros? São 13 cartas de ouro (evento) e um total de 52 cartas (espaço amostral).

$$P(\text{ouro}) = 13 / 52 = 25\%$$

6 Diferenças entre probabilidade e estatística

- **Estatística:** É a área de conhecimento que estuda a coleta, representação e análise de dados.
- **Probabilidade:** Ela é uma parte da estatística que compreende o estudo de eventos aleatórios e incertos.

DIFERENCIAL DESSA FORMAÇÃO EM TÉCNICAS DE PROGRAMAÇÃO

Vamos aprender a utilizar os

principais comandos da

linguagem Python para

explicar

Estatística e Probabilidade

```
import matplotlib.pyplot as plt
# Dados para o eixo x (anos)
anos = [2010, 2011, 2012, 2013, 2014, 2015]

# Dados para o eixo y (receitas)
receitas = [5000, 8000, 6500, 9000, 7500, 10000]

# Criando o gráfico de linhas
plt.plot(anos, receitas)

# Configurando rótulos e título do gráfico
plt.xlabel('Ano')
plt.ylabel('Receitas (em dólares)')
plt.title('Receitas ao longo dos anos')

# Exibindo o gráfico
plt.show()
```



7 Técnicas de programação orientadas à objetos

- **Orientação à objetos:** organiza o código em torno de objetos, que são representações de entidades do mundo real. Os conceitos básicos da orientação a objetos são:
 - **Objetos:** São instâncias de uma classe, e são compostos por atributos e métodos.
 - Atributos: são características dos objetos
 - Métodos: são as ações que o objeto pode realizar.
 - **Classes:** São os modelos que dipificam o objeto.
 - Na classe teremos atributos e métodos do objeto
 - **Atributos:** São as variáveis de classe, ou seja, eles definem a estrutura de uma classe.
 - As variáveis são pequenos espaços de memória reservados para armazenar dados.

```
import matplotlib.pyplot as plt
# Dados para o eixo x (anos)
anos = [2010, 2011, 2012, 2013, 2014, 2015]

# Dados para o eixo y (receitas)
receitas = [5000, 8000, 6500, 9000, 7500, 10000]

# Criando o gráfico de linhas
plt.plot(anos, receitas)

# Configurando rótulos e título do gráfico
plt.xlabel('Ano')
plt.ylabel('Receitas (em dólares)')
plt.title('Receitas ao longo dos anos')

# Exibindo o gráfico
plt.show()
```

8 Introdução à linguagem Python e suas principais estruturas

- **Python:** linguagem de programação de fácil aprendizagem e muito utilizada para ciência de dados, *softwares* e *machine learning*. Através dessa linguagem, conseguimos produzir gráficos que apresentam os resultados das propabilidades extraídas da Estatística.

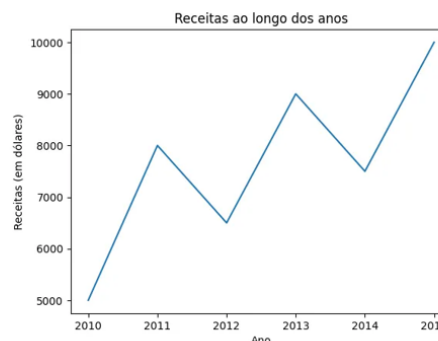
```
import matplotlib.pyplot as plt
# Dados para o eixo x (anos)
anos = [2010, 2011, 2012, 2013, 2014, 2015]

# Dados para o eixo y (receitas)
receitas = [5000, 8000, 6500, 9000, 7500, 10000]

# Criando o gráfico de linhas
plt.plot(anos, receitas)

# Configurando rótulos e título do gráfico
plt.xlabel('Ano')
plt.ylabel('Receitas (em dólares)')
plt.title('Receitas ao longo dos anos')

# Exibindo o gráfico
plt.show()
```



9 Exercícios

As resoluções serão apresentadas nessa formação

Como aprender a criar variáveis:

Tipo texto: `str (string)`

Tipo inteiro: `int`

Tipo número real: `float`

Tipo booleano: `bool`

Para executar os comandos em Python, vamos acessar o site <https://www.mycompiler.io/pt/new/python>

Vamos digitar os comandos a seguir para descobrir qual é o tipo das variáveis:

```
print (type("Alessandra"))
```

```
print (type("32"))
```

```
print*(type(1.71))
```

```
print*(type(true))
```


9 Exercícios

1. Converter as variáveis em outros tipos de dados:

```
altura = float("1.70") #converter texto para float
idade = int("24") #converter string para inteiro
altura_texto = str(1.70) #converter float em texto
print(altura, idade, altura_texto)
```

2. Um Dado possui o espaço amostral {1,2,3,4,5,6}

Se eu escolher o número 3, qual é o resultado: $P(3) = 1/6 \rightarrow 0,167$

3. Um dado é lançado 100 vezes. Se o número 4 aparecer 25 vezes, a probabilidade empírica de obter um 4 em um lançamento seria: $P(4) = 25/100 = 0,25$ ou 25%

9 Exercícios

4. Gráfico: Vamos representar as receitas (em dólares) ao longo de 6 anos

```
import matplotlib.pyplot as plt
# Dados para o eixo x (anos)
anos = [2010, 2011, 2012, 2013, 2014, 2015]

# Dados para o eixo y (receitas)
receitas = [5000, 8000, 6500, 9000, 7500, 10000]

# Criando o gráfico de linhas
plt.plot(anos, receitas)

# Configurando rótulos e título do gráfico
plt.xlabel('Ano')
plt.ylabel('Receitas (em dólares)')
plt.title('Receitas ao longo dos anos')

# Exibindo o gráfico
plt.show()
```



9 Exercícios

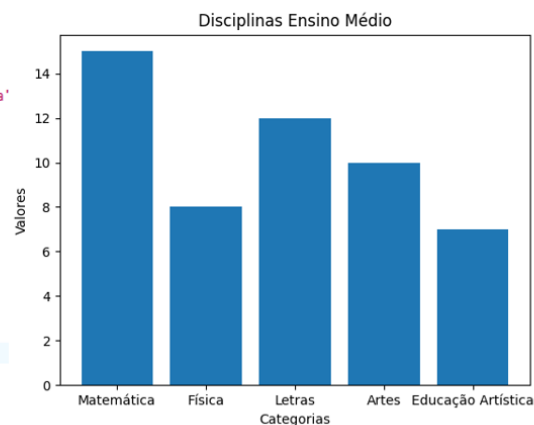
5. Gráfico: Vamos representar as disciplinas do Ensino Médio

```
import matplotlib.pyplot as plt
categorias = ['Matemática', 'Física', 'Letras', 'Artes', 'Educação Artística']
valores = [15, 8, 12, 10, 7]

# Criar o gráfico de barras
plt.bar(categorias, valores)

# Adicionar rótulos e título
plt.xlabel('Categorias')
plt.ylabel('Valores')
plt.title('Disciplinas Ensino Médio')

# Exibir o gráfico
plt.show()
```



9 Exercícios

6. Gráfico de pizza: proposta para a criação de um gráfico de distribuição de frutas

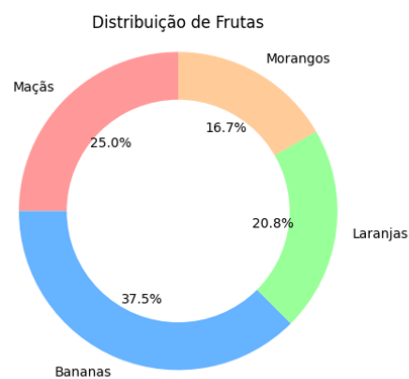
```
import matplotlib.pyplot as plt
# Dados para o gráfico de pizza
labels = ['Maçãs', 'Bananas', 'Laranjas', 'Morangos']
quantidades = [30, 45, 25, 20]
cores = ['#ff9999', '#66b3ff', '#99ff99', '#ffcc99']

# Criação do gráfico de pizza
plt.pie(quantidades, labels=labels, colors=cores,
        autopct='%1.1f%%', startangle=90)

# Adiciona um círculo no meio para transformar em um gráfico de pizza
centro_circulo = plt.Circle((0, 0), 0.70, fc='white')
fig = plt.gcf()
fig.gca().add_artist(centro_circulo)

# Ajustes estéticos
plt.axis('equal') # Para garantir que o gráfico seja exibido como um círculo
plt.title('Distribuição de Frutas')

# Mostra o gráfico de pizza
plt.show()
```



Referências

ESTATÍSTICA I. PROBABILIDADE. Aulas 2. **Professor Regina Meyer Branski**. Page 2. © 2010 Pearson Prentice Hall. Todos os direitos reservados.

Probabilidade. **Brasil Escola**. Disponível em: <https://brasilecola.uol.com.br/matematica/probabilidade.htm>>. Acesso em: 20 de out. de 2024.

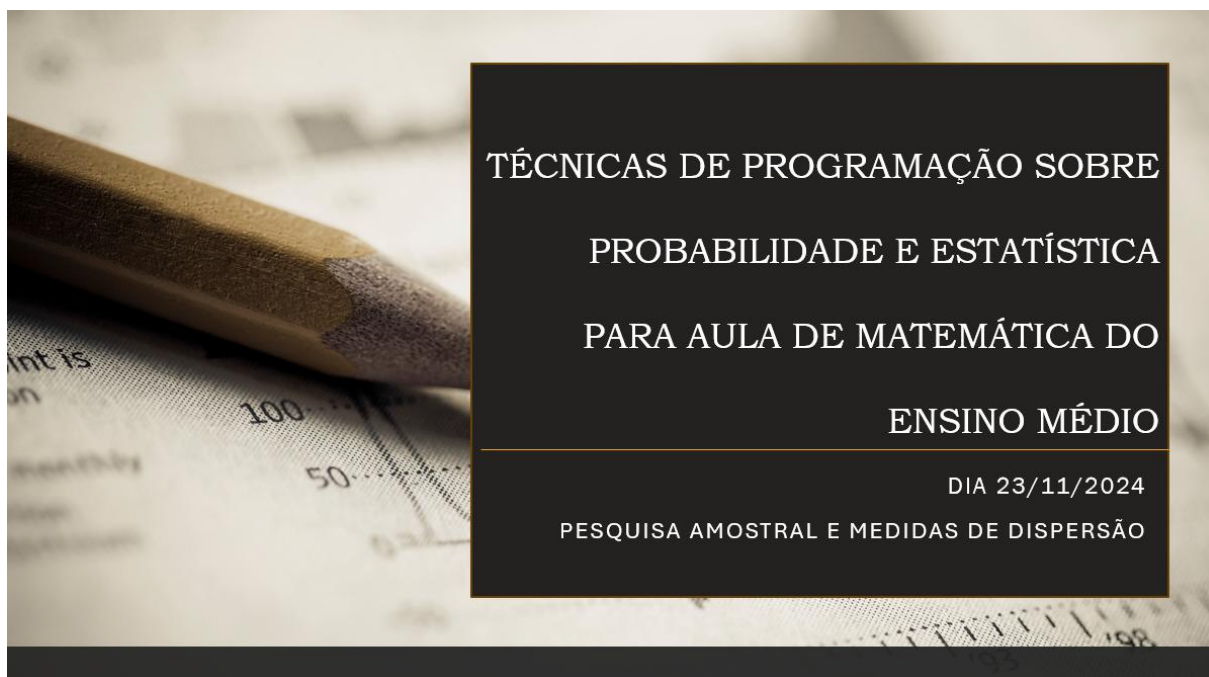
Lopes, C. E., & Meirelles, E. (2005). **O Desenvolvimento da Probabilidade e da Estatística**. *ENCONTRO REGIONAL DE PROFESSORES DE MATEMÁTICA*, 18, 1-8.

COSTA, N. P. L. O. Estatística. **São Paulo: Edgard Blücher**, 2002.

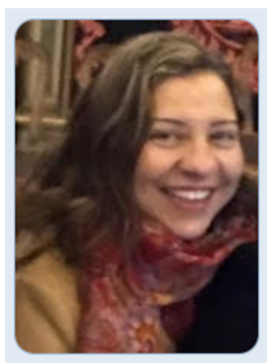
PIANA, C.F.B; MACHADO, A.A; SELAU, L.P.R. **Estatística Básica**. Departamento de Matemática e Estatística. 2013. (Apostila). Instituto de Física e Matemática, Universidade Federal de Pelotas. Disponível em: https://www.ufrgs.br/probabilidade-estatistica/extra/material/apostila_de_estatistica_basica.pdf. "Veja mais sobre "Probabilidade" em: <https://brasilecola.uol.com.br/matematica/probabilidade.htm>

APÊNDICE G – CONTEÚDO PROGRAMÁTICO DO 2º. ENCONTRO

Este apêndice tem o objetivo de apresentar o conteúdo programático compartilhado com os participantes do segundo encontro do evento de extensão intitulado: “técnicas de programação sobre Probabilidade e Estatística para aula de Matemática do ensino médio”.



Apresentação - Orientadora



Dra. Adriana Barroso de Azevedo

Possui graduação em Pedagogia pela Universidade Federal de Mato Grosso (1993), mestrado em Educação pela Universidade Federal de Mato Grosso (1997) e doutorado em Comunicação Social pela Universidade Metodista de São Paulo (2002), pós doutorado em Educação pela Universidade Federal do Rio Grande do Norte (2015). Atualmente é professora titular no PPGE da Universidade Metodista de São Paulo e professora convidada no PPGE da Universidade Municipal de São Caetano do Sul. É diretora Nacional de Educação da Rede Metodista de Educação. Tem experiência na área de Educação, é pesquisadora Narrativa atuando principalmente nos seguintes temas: educação a distância, educação básica, educação inclusiva, tecnologias digitais de informação e comunicação e epistemologia da pesquisa narrativa.

Apresentação

César Augusto do Prado Moraes



Possui graduação em Matemática pela Fundação Educacional de Penápolis (2004); Pedagogia pela Universidade Nove de Julho (2012); Letras - Português/Inglês(2019), História(2021) e Ciências Biológicas(2021) pelo Centro Universitário de Jales e Geografia pelo Centro Universitário FAVENI(2022); especializações em Metodologia do Ensino e Aprendizagem da Matemática pela Faculdade São Luís (2007) , Educação e Neurociências pela Faculdades Integradas Campos Salles (2019), Educação Inclusiva com Ênfase em Autismo pela Faculdade Casa Branca (2021) e Educação Matemática pela Faculdade Unida de São Paulo (2022), mestrado (2010) e doutorado (2018) em Educação pela Universidade Metodista de São Paulo e Pós-doutorado em Docência em Educação em Ciências e Matemática pela Universidade Federal do Pará (2021). Atualmente é professor universitário adjunto na Universidade Federal do Piauí, campus Bom Jesus- PI. Tem experiência na área de Matemática e Pedagogia, atuando principalmente nos seguintes temas: Estatística, Bioestatística, Matemática Financeira, Avaliação Educacional , Gestão Escolar , Metodologia do Ensino de Ciências e Matemática ,Tecnologia da Educação, História da Educação, Filosofia da Educação, Psicologia da Educação e Ética na Educação. Professor Colaborador do Programa de Pós-Graduação em Educação da Universidade Metodista de São Paulo na linha de Formação de Educadores.

Apresentação – Mestrando

Alessandra Soares dos Santos



Possui graduação em Análise e Desenvolvimento de Sistemas pela Faculdade Claretiano (2011), pós-graduação em Segurança da Informação pela Faculdade Impacta (2013), pós-graduação em Master em Information Systems em Arquitetura e Engenharia de Software pela Universidade Municipal de São Caetano do Sul (2023), mestrando em Educação pela Universidade Municipal de São Caetano do Sul. É produtora de material didático nas disciplinas de Tecnologia da Informação. Possui experiência em docência nas disciplinas de computação pelo Senac Vila Prudente, Codenation e Senai Santa Catarina. É Arquiteta de Tecnologia no Banco BV.

OBJETIVO DESTA FORMAÇÃO

Apresentar novas ferramentas didáticas para o ensino da Matemática e, principalmente, nos conteúdos de Estatística e Probabilidade

Aprender
linguagem
Phyton

Utilizar Phyton
para
resoluções de
Matemática

Utilizar essa
ferramenta
didática em
sala de aula

AGENDA

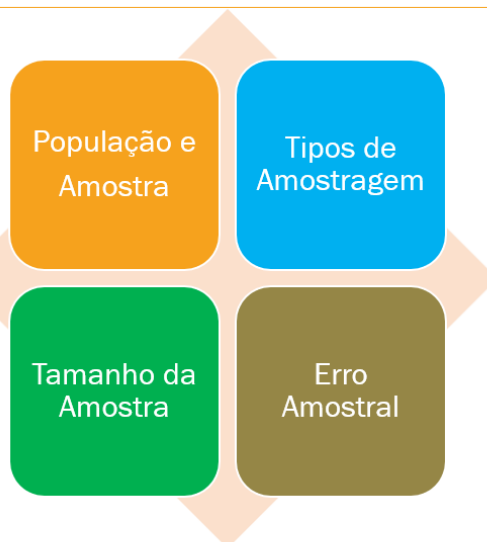
- 1 Conceitos básicos de amostragem
- 2 Tipos de variáveis: média, modal, e mediana
- 3 Medidas de dispersão: amplitude e desvio
- 4 Técnicas de programação orientada à objetos - Python
- 5 Exercícios

1 Conceitos básicos de amostragem

A amostragem é um processo estatístico essencial, especialmente quando trabalhamos com grandes populações.

Então, neste tópico vamos explicar alguns conceitos comuns sobre amostragem, são eles:

- População e Amostra
- Tipos de Amostragem
- Tamanho da Amostra
- Erro Amostral



1 Conceitos básicos de amostragem

1. População e Amostra

- **População:** Conjunto completo de itens ou pessoas sobre os quais queremos obter informações.
- **Amostra:** Subconjunto da população, selecionado para análise.

Exemplo: Pesquisa da estatura de uma escola com 90 alunos (**população: 90 alunos**) usando uma **amostra de 10%** da população:

1. numeram-se os alunos de 1 a 90;

2. sorteiam-se 9 números (10% de 90) usando algum **mecanismo aleatório** ou através de uma

Tabela de Números Aleatórios: 14 35 30 19 66 27 77 45 38

COMO É O MECÂNISMO
ALEATÓRIO EM PYTHON?

1 Conceitos básicos de amostragem

2. sorteiam-se 9 números (10% de 90) usando algum **mecanismo aleatório** ou através de uma

Tabela de Números Aleatórios: 14 35 30 19 66 27 77 45 38

COMO É O MECÂNISMO ALEATÓRIO EM PYTHON?

```
#biblioteca random permite trabalhar com números randômicos  
import random
```

```
#criar array  
espaco_amostral = [14, 35, 30, 19, 66, 27, 77, 45, 38]
```

```
#função choice é utilizada para  
numero_randomico = random.choice(espaco_amostral)
```

```
#exibir o número randômico  
print("número randômico:", numero_randomico)
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

Probabilística

Aleatória Simples

Sistemática

Estratificada

Conglomerados

Não-Probabilística

Conveniência

Intencional

Por Cotas

Bola de Neve

Desproporcional

1

Conceitos básicos de amostragem

2. Tipos de Amostragem

2.1 Probabilística:

- **Amostragem Aleatória Simples:** Cada elemento da população tem a mesma probabilidade de ser escolhido.

```
import numpy as np
import pandas as pd

# Lista de alunos
alunos = ['Ana', 'Bruno', 'Carlos', 'Diana', 'Eduardo', 'Fernanda', 'Gustavo', 'Helena', 'Igor', 'Julia']

# Criando um DataFrame com os alunos
dados_alunos = pd.DataFrame({'Nome': alunos})

# Definindo o tamanho da amostra
tamanho_amostra = 4

# Seleção de uma amostra aleatória simples
amostra_alunos = dados_alunos.sample(n=tamanho_amostra, random_state=1)

print("Alunos Originais:")
print(dados_alunos)
print("\nAmostra Aleatória Simples:")
print(amostra_alunos)
```

1

Conceitos básicos de amostragem

2. Tipos de Amostragem

2.1 Probabilística:

- **Amostragem Sistemática:** Quando os elementos da população já estão ordenados, não é necessário construir um sistema de referência ou de amostragem.

```
import numpy as np
#Tamanho da população e da amostra
populacao_tamanho = 1000
amostra_tamanho = 100

#Gerar uma população de exemplo
populacao = np.arange(populacao_tamanho)
#Calcular o intervalo (k) de amostragem
intervalo = populacao_tamanho // amostra_tamanho #divisão inteira

#Selecionar um ponto de partida aleatório entre 0 e o intervalo
ponto_de_partida = np.random.randint(0, intervalo)

#Selecionar a amostra usando amostragem sistemática
amostra_sistemica = populacao[ponto_de_partida::intervalo]
#operador:: indica a posição do índice

print("População:", populacao)
print("Amostra Sistemática:", amostra_sistemica)
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.1 Probabilística:

- **Amostragem Estratificada:**

A população é dividida em subgrupos (estratos), e amostras são retiradas de cada estrato.

```
import numpy as np
import pandas as pd

#Criação de um DataFrame de exemplo com estudantes e seus cursos
dados = pd.DataFrame({'ID': np.arange(1, 101),
                      'Curso': ['Engenharia'] * 30 +
                              ['Medicina'] * 40 + ['Direito'] * 30,
                      'Nota': np.random.uniform(0, 10, 100) })
#Notas variando de 0 a 10

#Tamanho da amostra desejada por curso
amostra_tamanhos = {'Engenharia': 10, 'Medicina': 15, 'Direito': 10}

#Amostragem estratificada manual
amostra = pd.DataFrame()

#Engenharia
estrato_engenharia = dados[dados['Curso'] == 'Engenharia']
amostra_engenharia = estrato_engenharia.sample(n=amostra_tamanhos['Engenharia'], random_state=42)
amostra = pd.concat([amostra, amostra_engenharia])
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.1 Probabilística:

- **Amostragem Estratificada:**

CONTINUAÇÃO DO CÓDIGO EM PYTHON

```
#Medicina
estrato_medicina = dados[dados['Curso'] == 'Medicina']
amostra_medicina = estrato_medicina.sample(n=amostra_tamanhos['Medicina'], random_state=42)
amostra = pd.concat([amostra, amostra_medicina])

#Direito
estrato_direito = dados[dados['Curso'] == 'Direito']
amostra_direito = estrato_direito.sample(n=amostra_tamanhos['Direito'], random_state=42)
amostra = pd.concat([amostra, amostra_direito])

print("Dados Originais:")
print(dados)
print("\nAmostra Estratificada:")
print(amostra)
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.1 Probabilística:

- **Amostragem por Conglomerados:** A população é dividida em grupos (conglomerados), e alguns grupos são selecionados aleatoriamente para análise completa.

```
import numpy as np
import pandas as pd

#Criação de um DataFrame de exemplo com estudantes agrupados em classes
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Classe': np.repeat(np.arange(1, 11), 10),
    #10 conglomerados (classes) com 10 estudantes cada
    'Nota': np.random.uniform(0, 10, 100)
    #Notas variando de 0 a 10
})

#Seleção de conglomerados (classes) aleatórios
np.random.seed(42)
conglomerados_selecionados = np.random.choice(dados['Classe'].unique(), size=3,
replace=False)

#Filtrando os dados para incluir apenas os conglomerados selecionados
amostra_conglomerados = dados[dados['Classe'].isin(conglomerados_selecionados)]

print("Dados Originais:")
print(dados)
print("\nConglomerados Selecionados:", conglomerados_selecionados)
print("Amostra por Conglomerados:")
print(amostra_conglomerados)
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.2 Não-Probabilística:

- **Amostragem por conveniência:** É uma técnica de amostragem em que os indivíduos são selecionados com base em sua disponibilidade e acessibilidade. Essa técnica é geralmente utilizada quando é difícil ou impossível obter uma amostra representativa da população de interesse.

```
import pandas as pd
import numpy as np

# Criando um DataFrame de exemplo com estudantes
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Nome': [f'Estudante {i}' for i in range(1, 101)],
    'Nota': np.random.uniform(0, 10, 100) # Notas variando de 0 a 10
})

# Seleção de uma amostra por conveniência (os primeiros 20 estudantes)
amostra_conveniencia = dados.head(20)

print("Dados Originais:")
print(dados)
print("\nAmostra por Conveniência:")
print(amostra_conveniencia)
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.2 Não-Probabilística:

- **Amostragem intencional:** É uma técnica de seleção de amostras em que a pessoa que está fazendo a pesquisa, escolhe cuidadosamente os indivíduos ou unidades de interesse com base em algum critério pré-determinado.

```
import pandas as pd
import numpy as np

# Criando um DataFrame de exemplo com estudantes
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Nome': [f'Estudante {i}' for i in range(1, 101)],
    'Nota': np.random.uniform(0, 10, 100) # Notas variando de 0 a 10
})

# Seleção de uma amostra intencional (estudantes com nota acima de 8)
amostra_intencional = dados[dados['Nota'] > 8]

print("Dados Originais:")
print(dados)
print("\nAmostra Intencional:")
print(amostra_intencional)
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.2 Não-Probabilística:

Amostragem por cotas: É uma técnica de seleção de amostras em que a pessoa que está fazendo a pesquisa define previamente certos critérios de inclusão para a escolha dos participantes da pesquisa.

```
import pandas as pd
import numpy as np

# Criando um DataFrame de exemplo com estudantes
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Curso': ['Engenharia'] * 30 + ['Medicina'] * 40 + ['Direito'] * 30,
    'Nota': np.random.uniform(0, 10, 100) # Notas variando de 0 a 10
})

# Definindo cotas: queremos 10 estudantes de Engenharia, 15 de Medicina e 5 de Direito
cota_engenharia = dados[dados['Curso'] == 'Engenharia'].sample(n=10, random_state=42)
cota_medicina = dados[dados['Curso'] == 'Medicina'].sample(n=15, random_state=42)
cota_direito = dados[dados['Curso'] == 'Direito'].sample(n=5, random_state=42)

# Concatenando as cotas para formar a amostra final
amostra_cotas = pd.concat([cota_engenharia, cota_medicina, cota_direito])

print("Dados Originais:")
print(dados)
print("\nAmostra por Cotas:")
print(amostra_cotas)
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.2 Não-Probabilística:

Amostragem bola de neve:

É uma técnica de seleção de amostras em que o pesquisador começa selecionando alguns participantes que se encaixam nos critérios de inclusão da pesquisa.

```
import pandas as pd
import numpy as np

#Lista inicial de participantes (semente)
participantes_iniciais = ['Participante 1', 'Participante 2']

#Dicionário simulando indicações de novos participantes
indicacoes = {
    'Participante 1': ['Participante 3', 'Participante 4'],
    'Participante 2': ['Participante 5', 'Participante 6'],
    'Participante 3': ['Participante 7', 'Participante 8'],
    'Participante 4': ['Participante 9', 'Participante 10']
}

#Etapa 1: Adicionando participantes iniciais
amostra_bola_de_neve = participantes_iniciais.copy()
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.2 Não-Probabilística:

Amostragem bola de neve:

CONTINUAÇÃO DO CÓDIGO EM PYTHON

```
#Etapa 2: Adicionando participantes indicados pelos iniciais
novos_participantes = []
novos_participantes.extend(indicacoes.get('Participante 1', []))
novos_participantes.extend(indicacoes.get('Participante 2', []))
amostra_bola_de_neve.extend(novos_participantes)

#Etapa 3: Adicionando participantes indicados pelos novos participantes
novos_participantes_temp = []
novos_participantes_temp.extend(indicacoes.get('Participante 3', []))
novos_participantes_temp.extend(indicacoes.get('Participante 4', []))
amostra_bola_de_neve.extend(novos_participantes_temp)

print("Amostra Final (Bola de Neve):")
print(amostra_bola_de_neve)
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.2 Não-Probabilística:

- **Amostragem desproporcional:** É uma técnica de seleção de amostras em que a pessoa que está fazendo a pesquisa escolhe deliberadamente um número diferente de participantes de cada grupo ou subgrupo da população, com base em um critério pré-determinado.

```
import pandas as pd
import numpy as np

# Criando um DataFrame de exemplo com diferentes grupos
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Grupo': ['Grupo A'] * 40 + ['Grupo B'] * 30 + ['Grupo C'] * 30,
    'Valor': np.random.uniform(0, 100, 100)
})

# Definindo os tamanhos desproporcionais para cada grupo
amostra_desproporcional = {
    'Grupo A': 10,
    'Grupo B': 15,
    'Grupo C': 5
}
```

1 Conceitos básicos de amostragem

2. Tipos de Amostragem

2.2 Não-Probabilística:

Amostragem desproporcional:

**CONTINUAÇÃO DO CÓDIGO EM
PYTHON**

```
#Selecionando amostras desproporcionais
amostra_A = dados[dados['Grupo'] == 'Grupo
A'].sample(n=amostra_desproporcional['Grupo A'], random_state=42)
amostra_B = dados[dados['Grupo'] == 'Grupo
B'].sample(n=amostra_desproporcional['Grupo B'], random_state=42)
amostra_C = dados[dados['Grupo'] == 'Grupo
C'].sample(n=amostra_desproporcional['Grupo C'], random_state=42)

#Concatenando as amostras para formar a amostra final
amostra_final = pd.concat([amostra_A, amostra_B, amostra_C])

print("Dados Originais:")
print(dados)
print("\nAmostra Desproporcional:")
print(amostra_final)
```

1 Conceitos básicos de amostragem

3. Tamanho da Amostra

O tamanho da amostra refere-se ao número de observações ou elementos que são selecionados a partir de uma população para compor uma amostra. A escolha do tamanho adequado da amostra é crucial para garantir a representatividade dos dados e a precisão das inferências estatísticas.

Importância do Tamanho da Amostra

- **Precisão:** Um tamanho de amostra maior tende a reduzir o erro amostral, proporcionando estimativas mais precisas da população.
- **Confiança:** Um tamanho adequado de amostra permite a construção de intervalos de confiança mais estreitos e conclusões mais confiáveis.
- **Variabilidade:** Levar em conta a variabilidade da população ajuda a determinar o tamanho da amostra necessário para capturar essa variabilidade de forma representativa.

```
import math
```

```
#Parâmetros para o cálculo do tamanho da amostra
Z = 1.96 #valor Z para 95% de nível de confiança
p = 0.5 #proporção da população
e = 0.05 #margem de erro
```

```
#Cálculo do tamanho da amostra
n = (Z**2 * p * (1 - p)) / (e**2)
```

```
#Arredondando para o próximo número inteiro
n = math.ceil(n)
```

```
print("Tamanho da Amostra:", n)
```

1 Conceitos básicos de amostragem

4. Erro Amostral

- **Erro de Amostragem:** Diferença entre as medidas da população e da amostra devido ao fato de a amostra ser apenas uma parte da população.

```
import numpy as np
import pandas as pd
```

```
# Gerar uma população de exemplo
np.random.seed(1)
populacao = np.random.randint(1, 3000, size=1000)
```

```
# Calcular a média da população
media_populacao = np.mean(populacao)
```

```
# Selecionar uma amostra aleatória
amostra = np.random.choice(populacao, size=100, replace=False)
```

```
# Calcular a média da amostra
media_amostra = np.mean(amostra)
```

1 Conceitos básicos de amostragem

4. Erro Amostral

- Erro de Amostragem:

CONTINUAÇÃO DO CÓDIGO EM PYTHON

```
# Calcular o erro de amostragem
erro_amostragem = media_amostra - media_populacao

print("Média da População:", media_populacao)
print("Média da Amostra:", media_amostra)
print("Erro de Amostragem:", erro_amostragem)
```

2 Tipos de variáveis: média, modal, e mediana

Tipos de variáveis

Média: Soma de todos os valores dividida pelo número de valores.

```
import numpy as np

# Lista de valores
valores = [10, 15, 10, 20, 30, 25, 10]

# Calcular a média
media = np.mean(valores)

print("Valores:", valores)
print("Média:", media)
```


2 Tipos de variáveis: média, modal, e mediana

Tipos de variáveis

Moda: Valor que mais se repete em um conjunto de dados.

```
import pandas as pd

#Lista de valores
valores = [10, 15, 10, 20, 30, 25, 10]

#Criar uma série do pandas
serie = pd.Series(valores)

#Calcular a moda
moda = serie.mode().values[0]

print("Valores:", valores)
Print("Série:", serie)
print("Moda:", moda)
```

2 Tipos de variáveis: média, modal, e mediana

Tipos de variáveis

Mediana: Valor central quando os dados são ordenados.

1. Ordenar os valores do array

2. Encontrar o valor central: Como há 5 valores, a mediana é o terceiro valor na lista ordenada, que é 20.

A mediana é particularmente útil em situações onde os dados podem conter outliers ou valores extremos, pois ela representa melhor o "valor típico" em tais casos.

```
import numpy as np

#Lista de valores
valores = [15, 20, 10, 30, 25]

#Ordenar os valores
valores_ordenados = sorted(valores)

#Calcular a mediana
mediana = np.median(valores)

print("Valores Ordenados:", valores_ordenados)
print("Mediana:", mediana)
```

3 Medidas de dispersão: amplitude e desvio

Medidas de dispersão

Amplitude: Essa medida de dispersão é definida como a diferença entre a maior e a menor observação de um conjunto de dados, isto é:

$$A = X_{\text{maior}} - X_{\text{menor}}$$

Por ser uma medida que não leva em consideração como os dados estão efetivamente distribuídos, não é muito utilizada.

```
import numpy as np

#Lista de valores
valores = [15, 20, 10, 30, 25]

#Calcular a amplitude
#Função np.max() para encontrar o valor máximo
#Função np.min() para encontrar o valor mínimo

amplitude = np.max(valores) - np.min(valores)

print("Valores:", valores)
print("Amplitude:", amplitude)
```

3 Medidas de dispersão: amplitude e desvio

Medidas de dispersão

Desvio: Em um conjunto de informações numéricas, o **desvio** é a “distância” de cada uma dessas informações até a **média aritmética** delas. Matematicamente, o desvio é obtido subtraindo cada um dos valores de um conjunto de informações da média aritmética desse conjunto.

```
import numpy as np

# Notas do aluno
notas = [1, 4, 9, 10]

# Calcular a média
media = np.mean(notas)

# Calcular o desvio padrão
desvio_padrao = np.std(notas, ddof=0) # ddof=0 para a população

print("Notas:", notas)
print("Média:", media)
print("Desvio Padrão:", desvio_padrao)
```

4 Técnicas de programação orientadas à objetos

- **Orientação à objetos:** organiza o código em torno de objetos, que são representações de entidades do mundo real. Os conceitos básicos da orientação a objetos são:
 - **Objetos:** São instâncias de uma classe, e são compostos por atributos e métodos.
 - **Atributos:** são características do objetos
 - **Métodos:** são as ações que o objeto pode realizar.
 - **Classes:** São os modelos que dipificam o objeto.
 - Na classe teremos atributos e métodos do objeto
 - **Atributos:** São as variáveis de classe, ou seja, eles definem a estrutura de uma classe.
 - As variáveis são pequenos espaços de memória reservados para armazenar dados.

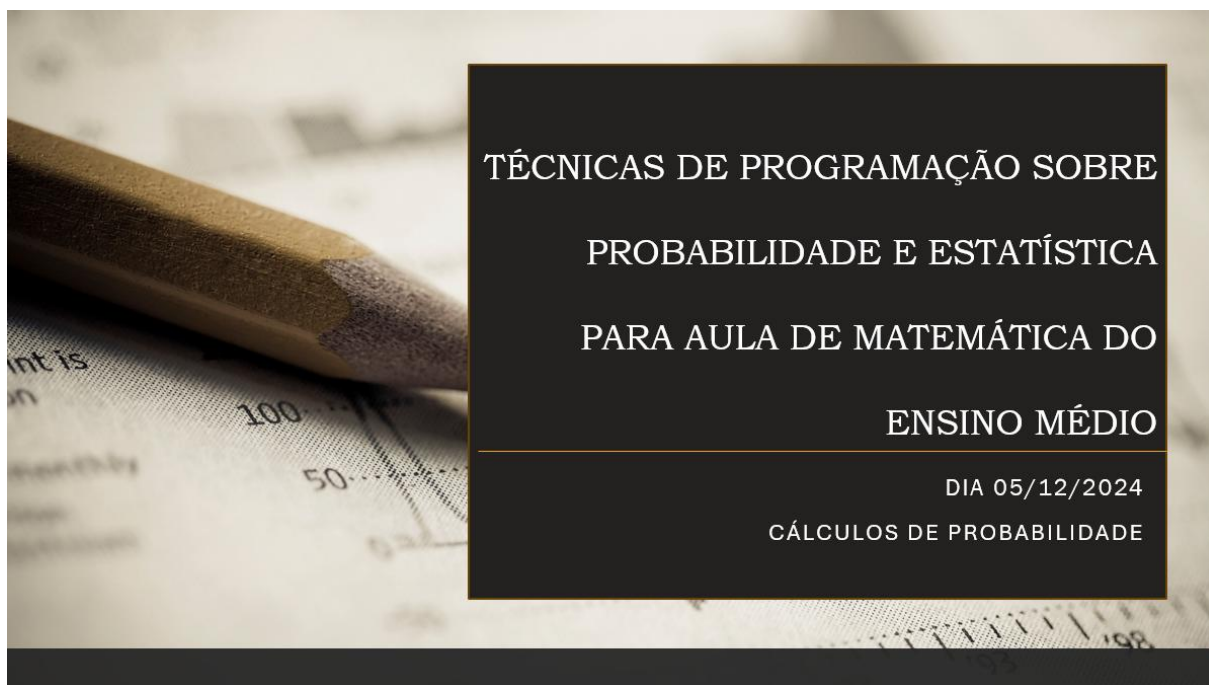
5 Exercícios

A lista de exercícios contempla os itens abordados em aula:

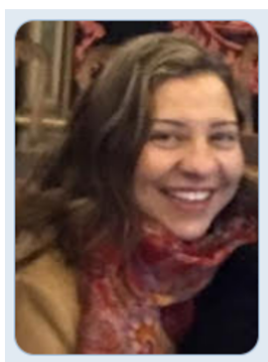
1. Conceitos básicos de amostragem
 1. População e Amostra
 2. Tipos de Amostragem
 3. Tamanho da Amostra
 4. Erro Amostral
2. Tipos de variáveis: média, modal, e mediana
3. Medidas de dispersão: amplitude e desvio

APÊNDICE H – CONTEÚDO PROGRAMÁTICO DO 3º. ENCONTRO

Este apêndice tem o objetivo de apresentar o conteúdo programático compartilhado com os participantes do terceiro encontro do evento de extensão intitulado: “técnicas de programação sobre Probabilidade e Estatística para aula de Matemática do ensino médio”.



Apresentação - Orientadora



Dra. Adriana Barroso de Azevedo

Possui graduação em Pedagogia pela Universidade Federal de Mato Grosso (1993), mestrado em Educação pela Universidade Federal de Mato Grosso (1997) e doutorado em Comunicação Social pela Universidade Metodista de São Paulo (2002), pós doutorado em Educação pela Universidade Federal do Rio Grande do Norte (2015). Atualmente é professora titular no PPGE da Universidade Metodista de São Paulo e professora convidada no PPGE da Universidade Municipal de São Caetano do Sul. É diretora Nacional de Educação da Rede Metodista de Educação. Tem experiência na área de Educação, é pesquisadora Narrativa atuando principalmente nos seguintes temas: educação a distância, educação básica, educação inclusiva, tecnologias digitais de informação e comunicação e epistemologia da pesquisa narrativa.

Apresentação

César Augusto do Prado Moraes



Possui graduação em Matemática pela Fundação Educacional de Penápolis (2004); Pedagogia pela Universidade Nove de Julho (2012); Letras - Português/Inglês(2019), História(2021) e Ciências Biológicas(2021) pelo Centro Universitário de Jales e Geografia pelo Centro Universitário FAVENI(2022); especializações em Metodologia do Ensino e Aprendizagem da Matemática pela Faculdade São Luís (2007) , Educação e Neurociências pela Faculdades Integradas Campos Salles (2019), Educação Inclusiva com Ênfase em Autismo pela Faculdade Casa Branca (2021) e Educação Matemática pela Faculdade Unida de São Paulo (2022), mestrado (2010) e doutorado (2018) em Educação pela Universidade Metodista de São Paulo e Pós-doutorado em Docência em Educação em Ciências e Matemática pela Universidade Federal do Pará (2021). Atualmente é professor universitário adjunto na Universidade Federal do Piauí, campus Bom Jesus- PI. Tem experiência na área de Matemática e Pedagogia, atuando principalmente nos seguintes temas: Estatística, Bioestatística, Matemática Financeira, Avaliação Educacional , Gestão Escolar , Metodologia do Ensino de Ciências e Matemática ,Tecnologia da Educação, História da Educação, Filosofia da Educação, Psicologia da Educação e Ética na Educação. Professor Colaborador do Programa de Pós-Graduação em Educação da Universidade Metodista de São Paulo na linha de Formação de Educadores.

Apresentação – Mestrando

Alessandra Soares dos Santos



Possui graduação em Análise e Desenvolvimento de Sistemas pela Faculdade Claretiano (2011), pós-graduação em Segurança da Informação pela Faculdade Impacta (2013), pós-graduação em Master em Information Systems em Arquitetura e Engenharia de Software pela Universidade Municipal de São Caetano do Sul (2023), mestrando em Educação pela Universidade Municipal de São Caetano do Sul. É produtora de material didático nas disciplinas de Tecnologia da Informação. Possui experiência em docência nas disciplinas de computação pelo Senac Vila Prudente, Codenation e Senai Santa Catarina. É Arquiteta de Tecnologia no Banco BV.

OBJETIVO DESTA FORMAÇÃO

Apresentar novas ferramentas didáticas para o ensino da Matemática e, principalmente, nos conteúdos de Estatística e Probabilidade

Aprender
linguagem
Phyton

Utilizar Phyton
para
resoluções de
Matemática

Utilizar essa
ferramenta
didática em
sala de aula

AGENDA

- 1 Usando espaços amostrais
- 2 Operações básicas de conjuntos
- 3 Aleatoriedade, probabilidade e simulação
- 4 Regras de adição e multiplicação
- 5 Probabilidade condicional e independência

1 Usando espaço amostrais

Ponto amostral: é uma das possíveis ocorrências ou resultados do experimento.

Exemplo:

Lançamento de um dado de seis faces:

➤ Espaço amostral: {1, 2, 3, 4, 5, 6}

➤ Qual é a probabilidade de obter o número 4?

```
import itertools
import random

# Espaço amostral
espaco_amostral = list(itertools.product([1,2,3,4,5,6], repeat=6))

# Simulação de um lançamento
resultado = random.choice(espaco_amostral)

print(f'Resultado do lançamento de duas moedas:
{resultado}')
```

2 Operações básicas de conjuntos

Operações básicas de conjuntos

União ($\cup$): Combinação de todos os elementos que pertencem a pelo menos um dos conjuntos.

Exemplo: Se $A = \{1, 2, 3\}$ e $B = \{3, 4, 5\}$, então
 $A \cup B = \{1, 2, 3, 4, 5\}$.

```
#Definindo os conjuntos
conjunto_a = {1, 2, 3}
conjunto_b = {3, 4, 5}

#União dos conjuntos
uniao = conjunto_a.union(conjunto_b)
print(f'União: {uniao}')
```

2 Operações básicas de conjuntos

Operações básicas de conjuntos

Interseção ($\cap$): Conjunto de todos os elementos que pertencem a ambos os conjuntos.

Exemplo: Se $A = \{1, 2, 3\}$ e $B = \{3, 4, 5\}$, então $A \cap B = \{3\}$.

```
# Definindo os conjuntos
conjunto_a = {1, 2, 3}
conjunto_b = {3, 4, 5}

# Interseção dos conjuntos
intersecao = conjunto_a.intersection(conjunto_b)
print(f"Interseção: {intersecao}")
```

2 Operações básicas de conjuntos

Operações básicas de conjuntos

Diferença ($\setminus$): Conjunto de todos os elementos que pertencem a um conjunto, mas não ao outro.

Exemplo: Se $A = \{1, 2, 3\}$ e $B = \{3, 4, 5\}$, então $A \setminus B = \{1, 2\}$ e $B \setminus A = \{4, 5\}$.

```
# Definindo os conjuntos
conjunto_a = {1, 2, 3}
conjunto_b = {3, 4, 5}

# Diferença dos conjuntos
diferenca_a_b = conjunto_a.difference(conjunto_b)
diferenca_b_a = conjunto_b.difference(conjunto_a)
print(f"Diferença A B: {diferenca_a_b}")
print(f"Diferença B A: {diferenca_b_a}")
```


2 Operações básicas de conjuntos

Operações básicas de conjuntos

Complemento: Conjunto de todos os elementos que não pertencem ao conjunto.

Definindo o conjunto e o conjunto universal

```
conjunto_a = {1, 2, 3}
```

```
universo = {1, 2, 3, 4, 5}
```

Exemplo: Se $A = \{1, 2, 3\}$ e o universo $U = \{1, 2, 3, 4, 5\}$, então o complemento de A é $A' = \{4, 5\}$.

Complemento do conjunto A

```
complemento_a = universo.difference(conjunto_a)
```

```
print(f'Complemento de A: {complemento_a}')
```

3 Aleatoriedade, probabilidade e simulação

Aleatoriedade

Refere-se à ocorrência de eventos de forma imprevisível e sem um padrão discernível.

Exemplo:

Lançamento de uma Moeda: Quando você lança uma moeda, o resultado (cara ou coroa) é imprevisível. Cada resultado tem uma probabilidade de 50%, mas não há como saber qual face será obtida em um único lançamento.

```
import random
```

Definindo o espaço amostral

```
espaco_amostral = ['Cara', 'Coroa']
```

Simulando o lançamento da moeda 1000 vezes

```
resultados = [random.choice(espaco_amostral) for _ in range(1000)]
```

Contando os resultados

```
caras = resultados.count('Cara')
```

```
coroas = resultados.count('Coroa')
```

Calculando as probabilidades

```
probabilidade_cara = caras / 1000
```

```
probabilidade_coroa = coroas / 1000
```

Exibindo os resultados

```
print(f"Resultados dos lançamentos: {caras} caras e {coroas} coroas")
```

```
print(f"Probabilidade de obter cara: {probabilidade_cara * 100}%")
```

```
print(f"Probabilidade de obter coroa: {probabilidade_coroa * 100}%")
```

3 Aleatoriedade, probabilidade e simulação

Probabilidade

É uma medida que quantifica a chance de um evento acontecer, expressa como um número entre 0 e 1, onde 0 indica que o evento nunca ocorre e 1 indica que o evento sempre ocorre.

```
import random

# Definindo o espaço amostral
espaco_amostral = ['Cara', 'Coroa']

# Simulando o lançamento da moeda
resultado = random.choice(espaco_amostral)
print(f'Resultado do lançamento da moeda: {resultado}')
```

Exemplo:

Lançamento de uma Moeda

- Espaço amostral: {Cara, Coroa}
- Probabilidade de obter "Cara":

3 Aleatoriedade, probabilidade e simulação

Simulação

É uma técnica usada para modelar e analisar o comportamento de sistemas complexos ou eventos aleatórios, replicando processos ou experimentos em um ambiente controlado.

```
import random

# Definindo o espaço amostral
espaco_amostral = [1, 2, 3, 4, 5, 6]

# Número de lançamentos
num_lancamentos = 10000

# Simulando os lançamentos do dado
resultados = random.choices(espaco_amostral, k=num_lancamentos)

# Calculando a frequência de cada resultado
frequencias = {face: resultados.count(face) for face in espaco_amostral}

# Calculando a probabilidade de cada resultado
probabilidades = {face: frequencia / num_lancamentos for face, frequencia in frequencias.items()}

# Exibindo os resultados
print(f'Frequências: {frequencias}')
print(f'Probabilidades: {probabilidades}')
```

4

Regras de adição e multiplicação

Regras de adição

É usada para calcular a probabilidade de que ocorra um de vários eventos mutuamente exclusivos (ou seja, eventos que não podem ocorrer ao mesmo tempo).

```
# Definindo as probabilidades individuais
P_A = 3/6 # Números pares: 2, 4, 6
P_B = 2/6 # Números maiores que 4: 5, 6
P_A_e_B = 1/6 # Número par maior que 4: 6
```

```
# Usando a regra da adição
```

```
P_A_ou_B = P_A + P_B - P_A_e_B
```

```
print(f"Probabilidade de obter um número par ou maior que
4: {P_A_ou_B}")
```

4

Regras de adição e multiplicação

Regras de multiplicação

É usada para calcular a probabilidade de que ambos os eventos ocorram juntos (eventos independentes).

```
# Definindo as probabilidades individuais
P_A = 1/2 # Probabilidade de obter "Cara"
P_B = 1/6 # Probabilidade de obter o número 4
```

```
# Usando a regra da multiplicação
```

```
P_A_e_B = P_A * P_B
```

```
print(f"Probabilidade de obter 'Cara' e o número 4:
{P_A_e_B}")
```

5

O que é probabilidade condicional e independência

Probabilidade condicional

É a probabilidade de um evento ocorrer dado que outro evento já ocorreu. Ela é representada por $P(A|B)$, que se lê como "a probabilidade de A dado B".

```
# Definindo as probabilidades
P_B = 1/4 # Probabilidade de a carta ser de Copas (13 de 52 cartas)
P_A_e_B = 1/52 # Probabilidade de a carta ser o Rei de Copas

# Calculando a probabilidade condicional
P_A_dado_B = P_A_e_B / P_B

print(f"Probabilidade de tirar o Rei de Copas dado que a carta é de Copas: {P_A_dado_B}")
```

5

O que é probabilidade condicional e independência

Probabilidade independência

Em termos de probabilidade significa que a ocorrência de um evento não afeta a ocorrência de outro evento.

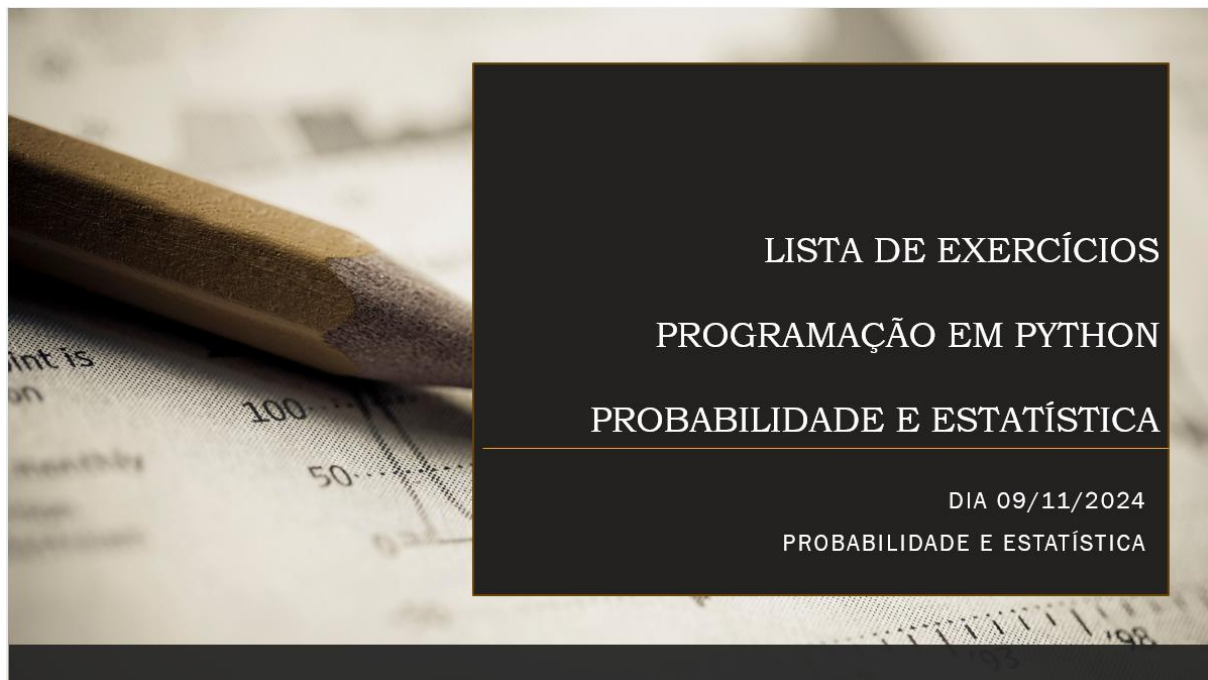
```
# Definindo as probabilidades
P_A = 1/2 # Probabilidade de obter "Cara"
P_B = 1/6 # Probabilidade de obter o número 4

# Calculando a probabilidade conjunta
P_A_e_B = P_A * P_B

print(f"Probabilidade de obter 'Cara' e o número 4: {P_A_e_B}")
```

APÊNDICE I – LISTA DE EXERCÍCIOS DO 1º. ENCONTRO

Este apêndice tem o objetivo de apresentar a lista de exercícios compartilhada com os participantes do primeiro encontro do evento de extensão intitulado: “técnicas de programação sobre Probabilidade e Estatística para aula de Matemática do ensino médio”.



Lista de Exercícios

A proposta dessa lista de exercícios é aplicarmos as técnicas de programação na linguagem Python para resolvermos questões de Probabilidade e Estatística.

Você pode utilizar as ferramentas abaixo para resolver os exercícios:

<https://www.mycompiler.io/pt/new/python>

VS Code – realizar o download: <https://code.visualstudio.com/download>



1 Exercícios

Proposta de resolução:

Em um saco com 692 M&Ms você encontrou 151 azuis.

Qual a porcentagem de M&M azuis?

$$151/692=0,218= 22\%$$

Se retirar um M&M qual a chance de que ele seja azul?

22%



Resultado do exercício na página seguinte

1 Exercícios

myCompiler

Introduza um título...

Python

```
1 import math
2
3 #criar a variável para o total de M&M
4 totalMm = 692
5
6 #criar variável apenas para os M&M na cor azul
7 totalMmAzul = 151
8
9 #criar variável para sabermos o total de M&M azuis
10 percentagemMmAzul = 151 / 692
11
12 #apresentar o resultado da variável percentagemMmAzul
13 print('Total de M&M azuis: ', percentagemMmAzul)
14
15 #Se retirar um M&M qual a chance de que ele seja azul?
16 print('Total de M&M azuis: ', percentagemMmAzul* 100)
17
18 #Observe que o resultado foi 21.820809248554912
19 #para arredondar as casas decimais, vamos utilizar a biblioteca math (importação realizada na linha 1)
20
21 #vamos criar a variável resultado para apresentarmos a porcentagem com valor arredondado
22 resultado = math.ceil(percentagemMmAzul * 100)
23
24 #vamos exibir a variável resultado
25 print(resultado, '%')
26
```

Entrada do programa

Saída do programa

Total de M&M azuis: 0.21820809248554912
Total de M&M azuis: 21.820809248554912
22 %

[Execution complete with exit code 0]

2 Exercícios

➤ **Clássica:** utilizada quando todos os resultados possíveis são igualmente prováveis e há um número finito de casos possíveis (resultado igualmente prováveis).

Fórmula: $P = \text{número casos favoráveis} / \text{número de casos possíveis}$

Proposta de Resolução: Um Dado possui o espaço amostral {1,2,3,4,5,6}.

Se eu escolher o número 3, qual é o resultado:

$$P(3) = 1/6 \rightarrow 0,167$$

```
1 import math
2
3 #criar a variável chamada dado - lembrando, essa variável é um array
4 dado = [1,2,3,4,5,6]
5
6 #o número 3 está representado no array apenas 1 vez, assim o nosso cálculo será:
7 resultado = 1/ 6
8
9 #vamos imprimir esse resultado
10 print(resultado)
11
12 #como aprendemos a trabalhar com arredondamento de valores, vamos aplicar nesse resultado
13 print(math.ceil(resultado * 100))
```

Entrada do programa

Saída do programa

0.16666666666666666

17

[Execution complete with exit code 0]

3 Exercícios

➤ **Empírica:** conhecida também como frequentista, é calculada a partir de repetições do experimento e análise dos resultados (baseada em uma pesquisa).

Fórmula: $P = \text{número de ocorrências do evento} / \text{total de observações}$

Proposta de Resolução: Um dado é lançado 100 vezes. Se o número 4 aparecer 25 vezes, a probabilidade empírica de obter um 4 em um lançamento seria:

$$P(4) = 25/100 = 0,25 \text{ ou } 25\%$$

```
1 import math
2
3 totallancamentoDado = 100
4
5 #total de vezes que aparece o número 4
6 totalVezes = 25
7
8 #como aprendemos a trabalhar com arredondamento de valores, vamos aplicar nesse resultado
9 print(math.ceil((totalVezes / totallancamentoDado) * 100))
```

Entrada do programa

Saída do programa

25

[Execution complete with exit code 0]

4 Tipos de probabilidade

➤ **Subjetiva:** resulta da intuição, de suposições fundamentadas e estimativas (palpite).

Fórmula: $P = \text{proposta do evento ocorrer} / \text{total de observações}$

Proposta de Resolução: Em uma pesquisa Gallup (empresa americana de pesquisa de opiniões), entre 855 adultos escolhidos aleatoriamente, 710 confirmaram ter voado em algum avião comercial. Como representar essa probabilidade:

P (voar em avião comercial) = 710 / 855 → 0,83%

Python

```
1 import math
2
3 adultos = 855
4
5 jaViaJaramVooComercial = 710
6
7 #como aprendemos a trabalhar com arredondamento de valores, vamos aplicar nesse resultado
8 print(math.ceil((jaViaJaramVooComercial / adultos) * 100), "%")
```

Entrada do programa

Saída do programa

84 %

[Execution complete with exit code 0]

5 Exercícios

➤ Em um lançamento de dados, qual a chance de sair um número par?

Proposta de Resolução: Um Dado possui o espaço amostral {1,2,3,4,5,6}.

$P(\text{par}) = 3/6 \rightarrow 0,5$

```
1 import math
2
3 #vamos criar uma variável para o espaço amostral de um dado, lembrando que será do tipo array
4 dado = [1,2,3,4,5,6]
5
6 chanceNumeroPar = 3
7
8 #a função len() é utilizada para sabermos o total de elementos em um array
9 totalElementosDado = len(dado)
10
11 #como aprendemos a trabalhar com arredondamento de valores, vamos aplicar nesse resultado
12 print(math.ceil((chanceNumeroPar / totalElementosDado) * 100), "%")
```

Entrada do programa

Saída do programa

50 %

[Execution complete with exit code 0]

6 Exercícios

Para representarmos o total de receitas ao longo dos anos, vou aprender como criar esse resultado utilizando gráfico. Para essa resolução fique à vontade para definir a quantidade de anos e valores.

Objetivo principal deste exercício é aprendermos a utilizar gráfico, usando a biblioteca matplotlib

```
import matplotlib.pyplot as plt
# Dados para o eixo x (anos)
anos = [2010, 2011, 2012, 2013, 2014, 2015]

# Dados para o eixo y (receitas)
receitas = [5000, 8000, 6500, 9000, 7500, 10000]

# Criando o gráfico de linhas
plt.plot(anos, receitas)

# Configurando rótulos e título do gráfico
plt.xlabel('Ano')
plt.ylabel('Receitas (em dólares)')
plt.title('Receitas ao longo dos anos')

# Exibindo o gráfico
plt.show()
```



7 Exercícios

Para representarmos as notas de um aluno nas disciplinas atuais, vamos utilizar o gráfico de barras para essa resolução.

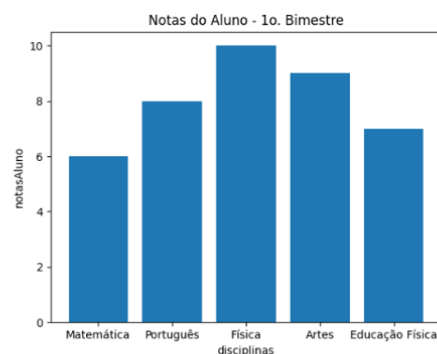
```
import matplotlib.pyplot as plt

disciplinas = ['Matemática', 'Português', 'Física', 'Artes', 'Educação Física']
notasAluno = [6, 8, 10, 9, 7]

plt.bar(disciplinas, notasAluno)

plt.xlabel('disciplinas')
plt.ylabel('notasAluno')
plt.title('Notas do Aluno - 1o. Bimestre')

plt.show()
```



8 Exercícios

Esse exercício é a mesma proposta do anterior, porém vamos utilizar outro modelo de gráfico

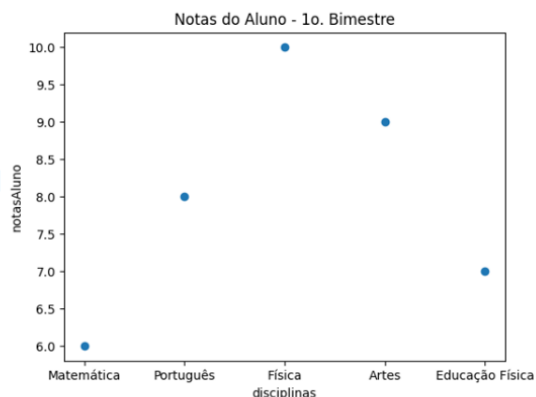
```
import matplotlib.pyplot as plt

disciplinas = ['Matemática', 'Português', 'Física', 'Artes', 'Educação Física']
notasAluno = [6, 8, 10, 9, 7]

plt.scatter(disciplinas, notasAluno)

plt.xlabel('disciplinas')
plt.ylabel('notasAluno')
plt.title('Notas do Aluno - 1o. Bimestre')

plt.show()
```



9 Exercícios

Para representarmos a quantidade de frutas, vamos utilizar o gráfico de pizza:

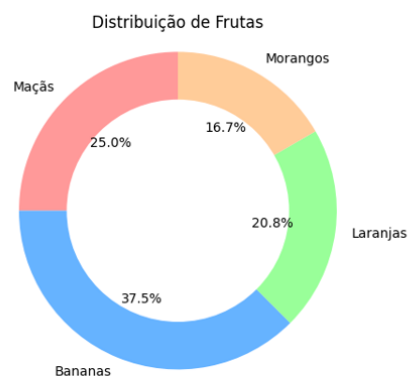
```
import matplotlib.pyplot as plt
# Dados para o gráfico de pizza
labels = ['Maçãs', 'Bananas', 'Laranjas', 'Morangos']
quantidades = [30, 45, 25, 20]
cores = ['#ff9999', '#66b3ff', '#99ff99', '#ffcc99']

# Criação do gráfico de pizza
plt.pie(quantidades, labels=labels, colors=cores,
       autopct='%1.1f%%', startangle=90)

# Adiciona um círculo no meio para transformar em um gráfico de pizza
centro_circulo = plt.Circle((0, 0), 0.70, fc='white')
fig = plt.gcf()
fig.gca().add_artist(centro_circulo)

# Ajustes estéticos
plt.axis('equal') # Para garantir que o gráfico seja exibido como um círculo
plt.title('Distribuição de Frutas')

# Mostra o gráfico de pizza
plt.show()
```



10 Exercícios

Nesse exercício é proposta é apresentarmos o conceito de espaço amostral.

Para isso, utilizamos a biblioteca `random` e a função `choice`

```
# Importar a biblioteca random
import random

# Definir o espaço amostral
espaco_amostral = [1, 2, 3, 4, 5, 6]

# Realizar o experimento (lançamento do dado)
resultado = random.choice(espaco_amostral)

# Exibir o resultado
print("O resultado do experimento foi:", resultado)
```

Conteúdo complementar

- 1) Introdução de noções de probabilidade e estatística nas séries do ensino fundamental e médio:
DA SILVA, BRASÍLIO RICARDO CIRILLO. INTRODUÇÃO DE NOÇÕES DE PROBABILIDADE E ESTATÍSTICA NAS SÉRIES DO ENSINO FUNDAMENTAL E MÉDIO.
- 2) Probabilidade na escola:
BAYER, Arno et al. Probabilidade na escola. In: **Congresso Internacional de Ensino de Matemática**. 2005.
- 3) Noções de probabilidade:
MAGALHÃES, Marcos Nascimento; DE LIMA, Antônio Carlos Pedroso. **Noções de probabilidade e estatística**. São Paulo, SP, Brazil: Edusp, 2002.
- 4) Probabilidade e Estatística:
SPIEGEL, Murray R.; SCHILLER, John J.; SRINIVASAN, R. Alu. **Probabilidade e Estatística-: Coleção Schaum**. Bookman Editora, 2016.
- 5) Compreendendo probabilidade e estatística:
BRAGA, Luis Paulo Vieira. **Compreendendo probabilidade e estatística**. Editora E-papers, 2010.
- 6) Desenvolvimento Orientado a Objetos utilizando a Linguagem Python:
Fundação Bradesco. Link do curso: <https://www.ev.org.br/cursos/desenvolvimento-orientado-a-objetos-utilizando-a-linguagem-python#:~:text=O%20principal%20paradigma%20da%20linguagem,na%20produ%C3%A7%C3%A3o%20de%20grandes%20projetos>.
- 7) Programação Orientada a Objetos
CARVALHO, Victorio Albani de; TEIXEIRA, Giovany Frossard. Programação Orientada a Objetos. 2016.

APÊNDICE J – LISTA DE EXERCÍCIOS DO 2º. ENCONTRO

Este apêndice tem o objetivo de apresentar a lista de exercícios compartilhada com os participantes do segundo encontro do evento de extensão intitulado: “técnicas de programação sobre Probabilidade e Estatística para aula de Matemática do ensino médio”.

FORMAÇÃO EM TÉCNICAS DE PROGRAMAÇÃO SOBRE PROBABILIDADE E ESTATÍSTICA PARA AULA DE MATEMÁTICA DO ENSINO MÉDIO

Aula: 23/11/2024 - Pesquisa amostral e medidas de dispersão

Lista de exercícios em linguagem de programação Python

Observação: Para a execução dos códigos em Python, utilizar o interpretador:
<https://onecompiler.com/python/>

Conceitos básicos de amostragem

- 1) População e Amostra: Em uma pesquisa de uma escola de 90 alunos (população), sorteamos 9 números (10% e 90) usando o mecanismo aleatório.

Pergunta: Em Python, como criar uma tabela e exibir números aleatórios.

```
#biblioteca random permite trabalhar com números randômicos  
import random
```

```
#criar array  
espaco_amostral = [14, 35, 30, 19, 66, 27, 77, 45, 38]
```

```
#função choice é utilizada para  
numero_randomico = random.choice(espaco_amostral)
```

```
#exibir o número randômico  
print("número randômico:", numero_randomico)
```

Tipos de amostragem

- 1) Amostragem aleatória simples
 - a. Com base em uma lista de alunos, como podemos criar uma amostragem aleatória simples?

```

import numpy as np
import pandas as pd

# Lista de alunos
alunos = ['Ana', 'Bruno', 'Carlos', 'Diana', 'Eduardo', 'Fernanda',
          'Gustavo', 'Helena', 'Igor', 'Julia']

# Criando um DataFrame com os alunos
dados_alunos = pd.DataFrame({'Nome': alunos})

# Definindo o tamanho da amostra
tamanho_amostra = 4

# Seleção de uma amostra aleatória simples
amostra_alunos = dados_alunos.sample(n=tamanho_amostra,
                                     random_state=1)

print("Alunos Originais:")
print(dados_alunos)
print("\nAmostra Aleatória Simples:")
print(amostra_alunos)

```

2) Amostragem sistemática:

- a. Com base em uma população de 1.000 pessoas e amostra com 100 elementos, como calculamos a amostragem sistemática?

```

import numpy as np

#Tamanho da população e da amostra
populacao_tamanho = 1000
amostra_tamanho = 100

#Gerar uma população de exemplo
populacao = np.arange(populacao_tamanho)

#Calcular o intervalo (k) de amostragem
intervalo = populacao_tamanho // amostra_tamanho
          #divisão inteira

#Selecionar um ponto de partida aleatório entre 0 e o intervalo
ponto_de_partida = np.random.randint(0, intervalo)

#Selecionar a amostra usando amostragem sistemática
amostra_sistemica = populacao[ponto_de_partida:intervalo]
                  #operador:: indica a posição do índice

print("População:", populacao)
print("Amostra Sistemática:", amostra_sistemica)

```

3) Amostragem estratificada:

- a. Com base em uma tabela (chamada dados), contendo: Id, Curso e Notas, criar uma tabela manual utilizando a tabela dados como parâmetro e aplicar o conceito de amostragem estratificada:

```
import numpy as np
import pandas as pd

#Criação de um DataFrame de exemplo com estudantes e seus cursos
dados = pd.DataFrame({'ID': np.arange(1, 101),
                      'Curso': ['Engenharia'] * 30 +
                               ['Medicina'] * 40 + ['Direito'] * 30,
                      'Nota': np.random.uniform(0, 10, 100) })
                      #Notas variando de 0 a 10

#Tamanho da amostra desejada por curso
amostra_tamANHos = {'Engenharia': 10, 'Medicina': 15, 'Direito': 10}

#Amostragem estratificada manual
amostra = pd.DataFrame()

#Engenharia
estrato_engenharia = dados[dados['Curso'] == 'Engenharia']
amostra_engenharia = estrato_engenharia.sample
                        (n=amostra_tamANHos
                        ['Engenharia'], random_state=42)
amostra = pd.concat([amostra, amostra_engenharia])
#Medicina
estrato_medicina = dados[dados['Curso'] == 'Medicina']
amostra_medicina = estrato_medicina.sample
                        (n=amostra_tamANHos['Medicina'], random_state=42)
amostra = pd.concat([amostra, amostra_medicina])
#Direito
estrato_direito = dados[dados['Curso'] == 'Direito']
amostra_direito = estrato_direito.sample
                        (n=amostra_tamANHos['Direito'], random_state=42)
amostra = pd.concat([amostra, amostra_direito])

print("Dados Originais:")
print(dados)
print("\nAmostra Estratificada:")
print(amostra)
```

4) Amostragem por conglomerados:

- a. Com base em uma tabela de alunos e suas respectivas notas, como obter uma amostragem por conglomerados?

```
import numpy as np
import pandas as pd
```

#Criação de um DataFrame de exemplo com estudantes agrupados em classes

```
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Classe': np.repeat(np.arange(1, 11), 10), #10
                                     conglomerados (classes) com 10
                                     estudantes cada
    'Nota': np.random.uniform(0, 10, 100)
                                     #Notas variando de 0 a 10
})
```

```
#Seleção de conglomerados (classes) aleatórios np.random.seed(42)
conglomerados_selecionados = np.random.choice
    (dados['Classe'].unique(), size=3, replace=False)
```

```
#Filtrando os dados para incluir apenas os conglomerados selecionados
amostra_conglomerados = dados[dados['Classe'].isin
    (conglomerados_selecionados)]
```

```
print("Dados Originais:")
print(dados)
print("\nConglomerados Selecionados:",conglomerados_selecionados)
print("Amostra por Conglomerados:")
print(amostra_conglomerados)
```

5) Amostragem por conveniência:

- a. Com base em uma tabela de estudantes e suas respectivas notas, como obter uma amostragem por conveniência?

```
import pandas as pd
import numpy as np
```

```
#Criando um DataFrame de exemplo com estudantes
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Nome': [f'Estudante {i}' for i in range(1, 101)],
    'Nota': np.random.uniform(0, 10, 100)
                                     #Notas variando de 0 a 10
})
```

```
#Seleção de uma amostra por conveniência
    (os primeiros 20 estudantes)
```

```
amostra_conveniencia = dados.head(20)
```

```
print("Dados Originais:")
print(dados)
```

```
print("\nAmostra por Conveniência:")
print(amostra_conveniencia)
```

6) Amostragem intencional:

- a. Com base em uma tabela de estudantes e suas respectivas notas, como obter uma amostragem intencional?

```
import pandas as pd
import numpy as np

# Criando um DataFrame de exemplo com estudantes
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Nome': [f'Estudante {i}' for i in range(1, 101)],
    'Nota': np.random.uniform(0, 10, 100)
    #Notas variando de 0 a 10
})

#Seleção de uma amostra intencional
(estudantes com nota acima de 8)

amostra_intencional = dados[dados['Nota'] > 8]

print("Dados Originais:")
print(dados)
print("\nAmostra Intencional:")
print(amostra_intencional)
```

7) Amostragem por cotas:

- a. Com base em uma tabela de cursos e notas, como obter uma amostragem por cotas?

```
import pandas as pd
import numpy as np

#Criando um DataFrame de exemplo com estudantes
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Curso': ['Engenharia'] * 30 + ['Medicina'] * 40 + ['Direito'] * 30,
    'Nota': np.random.uniform(0, 10, 100)
    #Notas variando de 0 a 10
})

#Definindo cotas: queremos 10 estudantes de Engenharia,
15 de Medicina e 5 de Direito

cota_engenharia = dados[dados['Curso'] == 'Engenharia'].
    sample(n=10, random_state=42)
cota_medicina = dados[dados['Curso'] == 'Medicina'].
    sample(n=15, random_state=42)
cota_direito = dados[dados['Curso'] == 'Direito'].
```



```

sample(n=5, random_state=42)

#Concatenando as cotas para formar a amostra final
amostra_cotas = pd.concat([cota_engenharia, cota_medicina,
                           cota_direito])

print("Dados Originais:")
print(dados)
print("\nAmostra por Cotas:")
print(amostra_cotas)

```

8) Amostragem bola de neve:

- a. Com base em um dicionário com 10 participantes, como obter uma amostragem bola de neve?

```

import pandas as pd
import numpy as np

#Lista inicial de participantes (semente)
participantes_iniciais = ['Participante 1', 'Participante 2']

#Dicionário simulando indicações de novos participantes
indicacoes = {
    'Participante 1': ['Participante 3', 'Participante 4'],
    'Participante 2': ['Participante 5', 'Participante 6'],
    'Participante 3': ['Participante 7', 'Participante 8'],
    'Participante 4': ['Participante 9', 'Participante 10']
}

#Etapa 1: Adicionando participantes iniciais

amostra_bola_de_neve = participantes_iniciais.copy()

#Etapa 2: Adicionando participantes indicados pelos iniciais

novos_participantes = []
novos_participantes.extend(indicacoes.get('Participante 1', []))
novos_participantes.extend(indicacoes.get('Participante 2', []))
amostra_bola_de_neve.extend(novos_participantes)

#Etapa 3: Adicionando participantes indicados pelos novos
participantes

novos_participantes_temp = []
novos_participantes_temp.extend(indicacoes.get('Participante 3', []))
novos_participantes_temp.extend(indicacoes.get('Participante 4', []))
amostra_bola_de_neve.extend(novos_participantes_temp)
print("Amostra Final (Bola de Neve):")
print(amostra_bola_de_neve)

```

9) Amostragem desproporcional:

- a. A tabela Grupos possui os campos: Id, Grupo e Valor. Com base no conceito de amostragem desproporcional, como podemos criar um código em Python utilizando a tabela mencionada?]

```
import pandas as pd
import numpy as np
```

```
# Criando um DataFrame de exemplo com diferentes grupos
```

```
dados = pd.DataFrame({
    'ID': np.arange(1, 101),
    'Grupo': ['Grupo A'] * 40 + ['Grupo B'] * 30 + ['Grupo C'] * 30,
    'Valor': np.random.uniform(0, 100, 100)
})
```

```
# Definindo os tamanhos desproporcionais para cada grupo
```

```
amostra_desproporcional = {
    'Grupo A': 10,
    'Grupo B': 15,
    'Grupo C': 5
}
```

```
#Selecionando amostras desproporcionais
```

```
amostra_A = dados[dados['Grupo'] == 'Grupo A'].sample(
    n=amostra_desproporcional['Grupo A'],
    random_state=42)
amostra_B = dados[dados['Grupo'] == 'Grupo B'].sample(
    n=amostra_desproporcional['Grupo B'],
    random_state=42)
amostra_C = dados[dados['Grupo'] == 'Grupo C'].sample(
    n=amostra_desproporcional['Grupo C'],
    random_state=42)
```

```
#Concatenando as amostras para formar a amostra final
```

```
amostra_final = pd.concat([amostra_A, amostra_B, amostra_C])
```

```
print("Dados Originais:")
print(dados)
print("\nAmostra Desproporcional:")
print(amostra_final)
```

Tamanho da amostra

A fórmula para calcular o tamanho da amostra depende de alguns fatores, como o nível de confiança desejado, a margem de erro aceitável e a variabilidade da população. A fórmula comum para amostras aleatórias simples é:

$$n = \frac{Z^2 \cdot p \cdot (1 - p)}{e^2}$$

Em Python, a exponenciação é representada por: **. O código para representarmos a fórmula acima é:

$$n = (Z^{**2} * p * (1 - p)) / (e^{**2})$$

1) Para cálculo de Tamanho da Amostra, temos os seguintes parâmetros:

- a. Z = 1.96 (95% do nível de confiança)
- b. P = 0.5 (proporção da população)
- c. E = 0.05 (margem de erro)

Com base nesses parâmetros, desenvolva um código em Python para representar o Tamanho da Amostra.

```
import math

#Parâmetros para o cálculo do tamanho da amostra
Z = 1.96 #valor Z para 95% de nível de confiança
p = 0.5 #proporção da população
e = 0.05 #margem de erro

#Cálculo do tamanho da amostra
n = (Z**2 * p * (1 - p)) / (e**2)

#Arredondando para o próximo número inteiro
n = math.ceil(n)

print("Tamanho da Amostra:", n)
```

Erro Amostral

Para representar o conceito de Erro Amostral em Python, utilizamos as seguintes funções:

A função **random.seed()** em Python é usada para inicializar o gerador de números aleatórios. Quando você define a semente (ou seed), você garante que a sequência de números aleatórios gerados será a mesma toda vez que você executar o código. Isso é muito útil em várias situações, como quando você deseja que seus

resultados sejam reprodutíveis para fins de teste, depuração, ou para compartilhar com outras pessoas que possam querer replicar seus resultados.

Como Funciona `random.seed()`: Quando você define uma semente usando `random.seed(n)`, você está dizendo ao gerador de números aleatórios para iniciar em um ponto específico na sequência de números pseudoaleatórios. O valor de `n` pode ser qualquer número inteiro.

A função **`random.randint`** faz parte do módulo `random` em Python e é usada para gerar números inteiros aleatórios dentro de um intervalo especificado. A sintaxe básica da função é: `random.randint(a,b, size= n)`, onde o parâmetro **`size`** cria um array conforme o tamanho informado.

- 1) Com base no conceito Erro Amostral, vamos criar um código em Python para gerar uma população, utilizando a função `Random.randint(1, 3000, size= 1000)` e realizar os seguintes cálculos:
 - a. Calcular a média da população: utilizar a função `mean(a)`
 - b. Selecionar uma amostra aleatória: utilizar a função `random.choice(a, size=100, replace=false)`
 - c. Calcular a média da amostra: utilizar a função `mean(a)`
 - d. Calcular o erro da amostragem: utilizar a fórmula: `erro_amostragem = media_amostra – media_população`

```
import numpy as np
import pandas as pd
```

```
#Gerar uma população de exemplo
np.random.seed(1)
populacao = np.random.randint(1, 3000, size=1000)
```

```
#Calcular a média da população
media_populacao = np.mean(populacao)
```

```
#Selecionar uma amostra aleatória
amostra = np.random.choice(populacao, size=100, replace=False)
```

```
#Calcular a média da amostra
media_amostra = np.mean(amostra)
```

```
#Calcular o erro de amostragem
erro_amostragem = media_amostra - media_populacao
```

```
print("Média da População:", media_populacao)
print("Média da Amostra:", media_amostra)
print("Erro de Amostragem:", erro_amostragem)
```

Tipos de Variáveis

1. **Média:** Para calcular a Média em Python, utilizamos a sequência:

- a. Criar um array de valores inteiros
- b. Utilizar a função `mean()` – Importar a biblioteca **numpy**

```
import numpy as np

# Lista de valores
valores = [10, 15, 10, 20, 30, 25, 10]

# Calcular a média
media = np.mean(valores)

print("Valores:", valores)
print("Média:", media)
```

2. **Moda:** Para calcular a Moda em Python, utilizamos a sequência:

- a. Criar um array de valores inteiros
- b. Utilizar a função `Series(n)` – Importar a biblioteca **pandas**
- c. Calcular a Moda utilizando a função: `moda = serie.mode().values[0]`

```
import pandas as pd

#Lista de valores
valores = [10, 15, 10, 20, 30, 25, 10]

#Criar uma série do pandas
serie = pd.Series(valores)

#Calcular a moda
moda = serie.mode().values[0]

print("Valores:", valores)
Print("Série:", serie)
print("Moda:", moda)
```

3. **Mediana:** Para calcular Mediana em Python, utilizamos a sequência:

- a. Criar um array de valores inteiros
- b. Ordenar o array utilizando a função `sorted(a)`
- c. Calcular a Mediana utilizando a função `median(a)` – Importar a biblioteca **numpy**

```
import numpy as np

#Lista de valores
valores = [15, 20, 10, 30, 25]

#Ordenar os valores
valores_ordenados = sorted(valores)
```

```
#Calcular a mediana
mediana = np.median(valores)

print("Valores Ordenados:", valores_ordenados)
print("Mediana:", mediana)
```

Medidas de dispersão: amplitude e desvio

- 1) Para representar o conceito de Medidas de dispersão – **Amplitude** em Python, utilizamos as seguintes funções:
 - a. `max()` – utilizada para encontrar o valor máximo de um array de valores
 - b. `min()` – utilizada para encontrar o valor mínimo de um array de valores
 - c. Para calcular a amplitude, utilizamos a fórmula: $\text{resultado} = \text{max}(a) - \text{min}(b)$

Com base no conceito de Medidas de dispersão, vamos criar um código em Python para representar a **Amplitude**, com os seguintes parâmetros:

 - Criar um array de valores inteiros
 - utilizar as `max(a)` e `min(n)` para encontrar ao valores máximo e mínimo do array – Importar a biblioteca **numpy**

```
import numpy as np

#Lista de valores
valores = [15, 20, 10, 30, 25]

#Calcular a amplitude
#Função np.max() para encontrar o valor máximo
#Função np.min() para encontrar o valor mínimo
amplitude = np.max(valores) - np.min(valores)

print("Valores:", valores)
print("Amplitude:", amplitude)
```

- 2) Para representar o conceito de Medidas de dispersão – **Desvio** em Python, utilizamos as seguintes funções:
 - a. `mean()` – utilizada para calcular a média de um array de valores – Importar a biblioteca **numpy**
 - b. `std(nome_variavel_array, ddof=0)`

A função `np.std` em Python, que faz parte da biblioteca **numpy**, é usada para calcular o desvio padrão ao longo de um eixo especificado de um array. O desvio padrão é uma medida de dispersão que indica o quão espalhados os valores de um conjunto de dados estão em torno da média.

Com base no conceito de Medidas de dispersão, vamos criar um código em Python para representar a **Desvio**, com os seguintes parâmetros:

 - Criar um array de valores inteiros
 - Calcular a média – utilizar a função `mean()`
 - Calcular o desvio padrão – utilizar a função: `desvio_padrao = np.std(notas, ddof=0)`

```
import numpy as np

#Notas do aluno
notas = [1, 4, 9, 10]

#Calcular a média
media = np.mean(notas)

#Calcular o desvio padrão
desvio_padrao = np.std(notas, ddof=0) # ddof=0 para a população

print("Notas:", notas)
print("Média:", media)
```

APÊNDICE K – NARRATIVAS DOCENTES

Este apêndice tem o objetivo de apresentar as narrativas dos docentes participantes do evento de extensão intitulado: “técnicas de programação sobre Probabilidade e Estatística para aula de Matemática do ensino médio”

Altas. (Docente 1)

"É muito complexo uma abordagem direta dos conteúdos, inicialmente as escolas poderiam fornecer ambientes que poderíamos aprofundar o conteúdo. Aqui na escola que leciono na PB não existe esse tipo de ambiente. Fica aqui o meu agradecimento a todos que fazem parte desse projeto. É algo inovador e incrível. " (Docente 2)

Adquirir cada vez mais conhecimento (Docente 5)

Novos conhecimentos (Docente 7)

Ampliar a minha didática com tecnologia de programação orientada à resolução de problemas de ensino médio. (Docente 8)

Poder elaborar atividades para apresentar e estimular meus alunos do ensino médio a usarem a programação nas aulas de matemática. (Docente 9)

Espero adquirir mais conhecimento e novas técnicas de ensino para poder passar para meus alunos. (Docente 10)

Minha expectativa é positiva para o conhecimento renovador. (Docente 12)

Tenho experiência muito básica com algoritmos e pensamento computacional. Estou curiosa, mas não sei muito bem o que esperar na prática. (Docente 14)

Explorar novas metodologias que integrem programação e Matemática, tornando o ensino mais interativo e visual. (Docente 17)

É interessante pois não tenho experiência na programação, então é algo que pode ser acrescentado a aula, para estimular alunos a usarem a ferramenta e despertar mais interesse na disciplina, no conteúdo. (Docente 19)
as expectativas de aprender programação, estatística e probabilidade podem ser desafiadoras, mas também extremamente gratificantes. À medida que você avança, você vê como esses conhecimentos se interligam e abrem novas oportunidades para resolver problemas complexo (Docente 21)

"Minha experiência nessa formação foi uma grande oportunidade, tanto no aspecto pedagógico quanto na ampliação de perspectivas sobre as práticas didáticas. Durante as três sessões, adquiri conhecimentos sobre a linguagem de programação que integram teoria e prática, além de implementar alguns testes por meio de metodologias ativas em meus momentos de sala de aula, que tornaram o aprendizado mais significativo para os estudantes. Essas experiências serão importantes para o ensino em sala de aula porque me forneceu ferramentas para diversificar minha prática pedagógica, tornando as aulas mais interativas e alinhadas às necessidades dos alunos. Concluo que me preparou para inovar no ensino dos conteúdos didáticos, promovendo um ambiente de aprendizagem mais dinâmico, criativo e significativo, onde os alunos possam não apenas aprender os conteúdos matemáticos de forma

mais integrada com a BNCC de Computação, mas também se envolver e se conectar com os conteúdos de forma profunda e contextualizada". (Docente 8)

Os exemplo dados foram para estatística, como dou aula para o Ensino Médio, acredito conseguir introduzir este tema nas minhas aulas. (Docente 9)

Eles irão ajudar no aprendizado da prática e ajudar meu aluno estratégias elaboradas. (Docente 12)

Os conteúdos apresentados no curso irão tornar as aulas de Estatística mais interativas e aplicáveis, utilizando Python para criar visualizações e simulações práticas. Isso facilitará a compreensão dos conceitos pelos alunos, tornando as aulas mais tecnológicas e atrativas, além de desenvolver habilidades importantes como lógica e análise de dados. (Docente 17)

Considerando o curso "FORMAÇÃO EM TÉCNICAS DE PROGRAMAÇÃO SOBRE PROBABILIDADE E ESTATÍSTICA PARA AULA DE MATEMÁTICA DO ENSINO MÉDIO" minha expectativa na formação era ser abrangente juntando o conteúdo de probabilidade e estatística com técnicas de programação, foram utilizados vários e bons os exemplos, porém pensei que pelo menos um ponto de cada habilidade seria explorado, mas o curto tempo do curso dificultou esta abrangência. Pensando na minha experiência e resposta dos alunos eu gostaria de ter mais exemplos do cotidiano dos alunos, desta forma o conteúdo se mostra útil e prático despertando a atenção trazendo o aluno para a aula e o conteúdo. O curso foi inspirador, despertou a necessidade de conhecer mais das técnicas de programação com mais estudo e prática para ser bem aplicado. A formação também serviu para mostrar e indicar uma direção para agregar mais conteúdo e possibilidade de aprendizagem nas aulas de matemática. Assim os pontos que foram destacados na formação podem ser explorados em sala, chamando a atenção dos alunos para o conteúdo obrigatório probabilidade e estatística, junto com a utilização de uma ferramenta prática, favorecendo o emprego da disciplina no dia a dia sem precisar lembrar de toda parte teórica, mas recordando dos comandos da programação, isto vai familiarizar o aluno com uma área de trabalho, que ele pode ter interesse profissional. Destaco algumas dificuldades no uso frequente da ferramenta em sala: não são todas escolas que possuem laboratório, computadores e, que estejam disponíveis as aulas de matemática; pode-se utilizar o celular mas vão surgir dificuldades técnicas como capacidade do celular, internet disponível, o aluno ter o aparelho, entre outros pontos. Neste ano temos a legislação dificultando o uso de celular em sala, assim por mais que se queira aproveitar, com o professor preparado para empregar a ferramenta, é difícil reunir todos os pontos favoráveis para utilizar técnicas de programação e ensinar probabilidade e estatística de uma forma mais amigável para o aluno. Uma sugestão era ter algum destaque das técnicas de programação disponíveis nos materiais que as escolas utilizam, assim provocaria professor e aluno para a utilizar ou no mínimo falar deste assunto. Agradeço à formação. (Docente 19)

ANEXO A – CERTIFICADO DO CURSO DE EXTENSÃO UNIVERSITÁRIA

	MINISTÉRIO DA EDUCAÇÃO UNIVERSIDADE FEDERAL DO PIAUÍ PRÓ-REITORIA DE EXTENSÃO E CULTURA COORDENADORIA DE PROGRAMAS, PROJETOS E EVENTOS CIENTÍFICOS E TECNOLÓGICOS/PREXC - CPPECT/PREXC	
CERTIFICADO		
Conferido a _____		
por haver participado do EVENTO DE EXTENSÃO "TÉCNICAS DE PROGRAMAÇÃO SOBRE PROBABILIDADE E ESTATÍSTICA PARA AULA DE MATEMÁTICA DO ENSINO MÉDIO", no período de 9 de Novembro de 2024 a 7 de Dezembro de 2024, perfazendo um total de 60 horas .		
Teresina, 12 de Dezembro de 2024		
Daniel Loucana da Costa Araújo Coordenador(a) da CPPECT/PREXC	Cesar Augusto do P. Moraes Coordenador(a) do(a) EVENTO	_____ Participante
Código de verificação: ec37e9404a Número do Documento: 785320 Identificador: 239179 Para verificar a autenticidade deste documento acesse http://www.sigaa.ufpi.br/sigaa/public, informando o Número do Documento, data de emissão do documento e o código de verificação.		

Sobre a formação: você irá aprender na prática como aplicar as técnicas de programação com o uso da linguagem Python, para os principais conteúdos de probabilidade e estatística da disciplina de matemática no ensino médio. São eles: probabilidade de estatísticas, pesquisa amostral (média, moda e mediana) e dispersão, por último, cálculos de probabilidades.

Objetivo: os professores, a partir desse curso, estarão aptos a desenvolver em suas aulas de Matemática, técnicas de programação para o ensino dos principais conteúdos de Probabilidade e Estatística.

Sobre a formação: Apresentar outras propostas de ferramentas didáticas no ensino de conteúdos de Probabilidade e Estatística da disciplina de Matemática, em consonância com as áreas de conhecimento previstas pela Base Nacional Comum Curricular do Ensino Médio (BNCC) para o ensino médio.