



TERMO DE CONDUTA

Declaro assumir o compromisso de confidencialidade e de sigilo escrito, fotográfico e verbal sobre as questões do exame ou avaliação pessoal que me serão apresentadas, durante o curso desta disciplina. Comprometo-me a não revelar, reproduzir, utilizar ou dar conhecimento, em hipótese alguma, a terceiros, e a não utilizar tais informações para gerar benefício próprio ou de terceiros. Reitero minha ciência de que não poderei fazer cópia manuscrita, registro fotográfico, filmar ou mesmo gravar os enunciados que me são apresentados.

Também me comprometo a não postar as soluções das questões em nenhum site ou arquivo na nuvem, bem como não as enviarei, para qualquer recipiente, seja por correio eletrônico, ou por qualquer outro meio disponível.

Declaro, ainda, estar ciente de que o não cumprimento de tais normas caracteriza infração ética, podendo acarretar punição de acordo com as regras da minha universidade.

Ciente,

Assinatura do aluno

Fundação CECIERJ - Vice Presidência de Educação Superior a Distância
Curso de Tecnologia em Sistemas de Computação
Disciplina: Programação com Interfaces Gráficas
APX1 1º semestre de 2022.

Nome -

Assinatura -

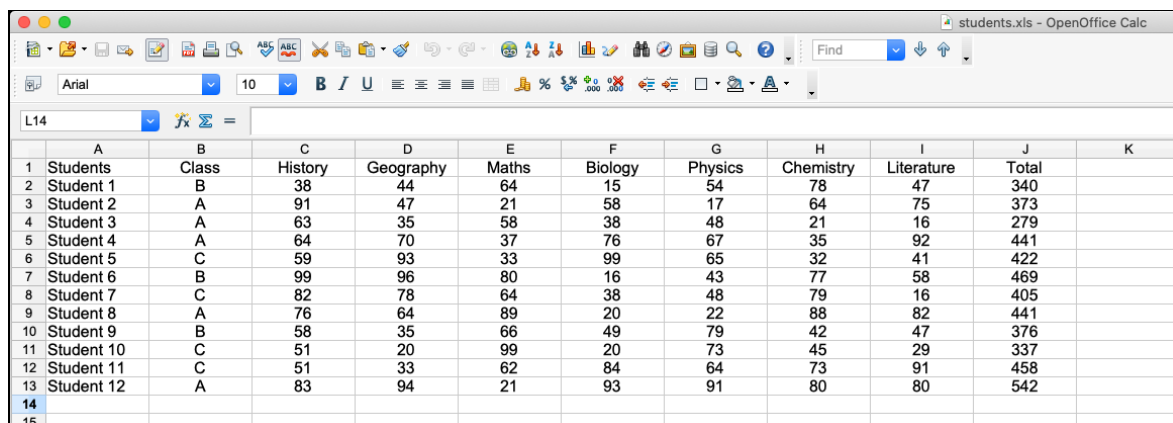
1. (10 pontos) Nessa AP1¹, você irá ler e processar uma planilha com notas de alunos de várias turmas.

*VBA*² foi incorporado ao Microsoft Excel 5.0 em 1993, e é uma linguagem de programação guiada por eventos (event-driven) que permite estender as aplicações do Office.

Por exemplo, com *VBA* é possível inserir ou remover linhas de uma planilha, bem como executar cálculos matemáticos para atualizar valores de um conjunto de células de uma tabela. No entanto, há quem diga que Pandas pode e irá substituir *VBA* por completo no processamento de planilhas.

A figura 1 exibe uma planilha com as notas das turmas A, B e C em 7 disciplinas que compõem um determinado curso.

Sua tarefa é ler e aplicar uma curva às notas obtidas por todos os alunos de uma dada disciplina, e implementar uma interface textual, para exibir as notas ajustadas, por exemplo, usando o método **apply** da classe `DataFrame`³. O código1 apresenta algumas funções do Pandas:



	A	B	C	D	E	F	G	H	I	J	K
	Students	Class	History	Geography	Maths	Biology	Physics	Chemistry	Literature	Total	
1	Student 1	B	38	44	64	15	54	78	47	340	
2	Student 2	A	91	47	21	58	17	64	75	373	
3	Student 3	A	63	35	58	38	48	21	16	279	
4	Student 4	A	64	70	37	76	67	35	92	441	
5	Student 5	C	59	93	33	99	65	32	41	422	
6	Student 6	B	99	96	80	16	43	77	58	469	
7	Student 7	C	82	78	64	38	48	79	16	405	
8	Student 8	A	76	64	89	20	22	88	82	441	
9	Student 9	B	58	35	66	49	79	42	47	376	
10	Student 10	C	51	20	99	20	73	45	29	337	
11	Student 11	C	51	33	62	84	64	73	91	458	
12	Student 12	A	83	94	21	93	91	80	80	542	
14											
15											

Figura 1: Planilha de notas das turmas A, B e C.

¹Requer Pandas >= 1.3 - <https://pandas.pydata.org>

²<https://www.techtudo.com.br/noticias/2016/03/o-que-e-e-como-usar-vba-no-excel.ghml>

³<https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.apply.html>

Código 1: Pandas

```
#!/usr/bin/env python
# coding: UTF-8
#
## @package data
#
# How to replace VBA with Python: OpenPyxl, Pandas, xlWings
# MacOS:
# - py310-xlrd
# - py310-openpyxl
#
# Ubuntu:
# - python3-xlrd
# - python3-openpyxl
#
# @author Paulo Roma
# @since 04/11/2021
#
import pandas as pd

# create a data frame
try:
    df = pd.read_excel(r'students.xls', sheet_name='Marks', engine="xlrd")
except ValueError:
    df = pd.read_excel( r'students.xlsx', sheet_name='Marks', engine="openpyxl")

print('The whole spreadsheet')
print(df)
print()

# filter Geography grades > 60
results = df.loc[df['Geography'] > 60]
print(df['Geography'] > 60)
print()
print('Geography > 60')
print(results)
print()

# all students in class A
results = df[df['Class'].str.match('A')]
print('Students in Class A')
print(results)
print()

# all students with a 2 digit: 2, 12, 22 ...
results = df[df['Students'].str.contains('2')]
print('Students with a 2 digit')
print(results)
print()

# group by Class
results = df.groupby('Class')

results = results.get_group("A").mean(numeric_only=True)
print('Mean Grouped by Class A')
print(results)
print()

# write to another workbook
results.to_excel(r'results.xlsx', sheet_name='results')
```

Eis a saída do código1:

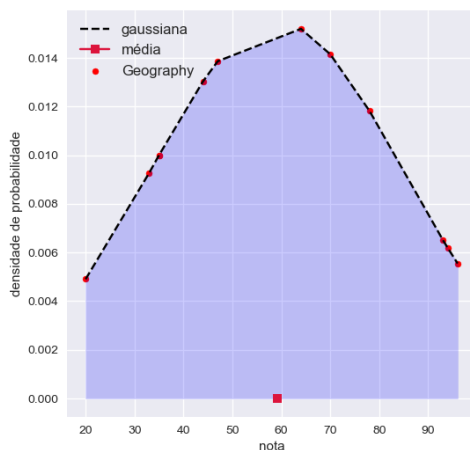
```
The whole spreadsheet
Students Class History Geography Maths Biology Physics Chemistry Literature Total
0 Student 1 B 38 44 64 15 54 78 47 340
1 Student 2 A 91 47 21 58 17 64 75 373
2 Student 3 A 63 35 58 38 48 21 16 279
3 Student 4 A 64 70 37 76 67 35 92 441
4 Student 5 C 59 93 33 99 65 32 41 422
5 Student 6 B 99 96 80 16 43 77 58 469
6 Student 7 C 82 78 64 38 48 79 16 405
7 Student 8 A 76 64 89 20 22 88 82 441
8 Student 9 B 58 35 66 49 79 42 47 1376
9 Student 10 C 51 20 99 20 73 45 29 337
10 Student 11 C 51 33 62 84 64 73 91 458
11 Student 12 A 83 94 21 93 91 80 80 542

Geography > 60
Students Class History Geography Maths Biology Physics Chemistry Literature Total
3 Student 4 A 64 70 37 76 67 35 92 441
4 Student 5 C 59 93 33 99 65 32 41 422
5 Student 6 B 99 96 80 16 43 77 58 469
6 Student 7 C 82 78 64 38 48 79 16 405
7 Student 8 A 76 64 89 20 22 88 82 441
11 Student 12 A 83 94 21 93 91 80 80 542

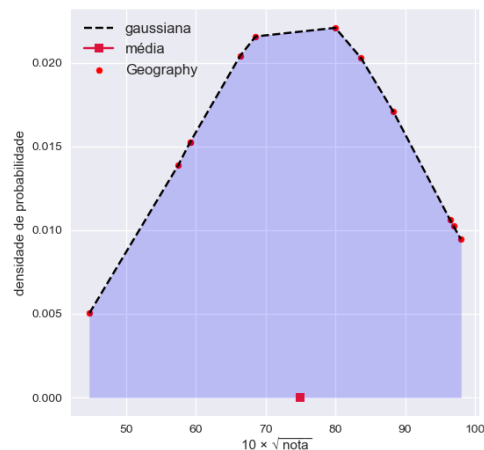
Students in Class A
Students Class History Geography Maths Biology Physics Chemistry Literature Total
1 Student 2 A 91 47 21 58 17 64 75 373
2 Student 3 A 63 35 58 38 48 21 16 279
3 Student 4 A 64 70 37 76 67 35 92 441
7 Student 8 A 76 64 89 20 22 88 82 441
11 Student 12 A 83 94 21 93 91 80 80 542

Students with a 2 digit
Students Class History Geography Maths Biology Physics Chemistry Literature Total
1 Student 2 A 91 47 21 58 17 64 75 373
11 Student 12 A 83 94 21 93 91 80 80 542

Mean Grouped by Class A
History 75.4
Geography 62.0
Maths 45.2
Biology 57.0
Physics 49.0
Chemistry 57.6
Literature 69.0
Total 415.2
dtype: float64
```



(a) Curva original de Geografia.



(b) Curva ajustada de Geo ($10\sqrt{\text{nota}}$).

Figura 2: Curvas das notas.

Para ajudá-lo nessa tarefa, existem instruções em vários sites ^{4 5} que explicam, detalhadamente, como ajustar um conjunto de notas. Há várias formas para se ajustar as notas de uma turma, e é função do professor escolher a mais adequada a cada situação: ^{6 7}

A sua interface deve permitir selecionar a curva a ser aplicada ⁸ às notas de uma disciplina

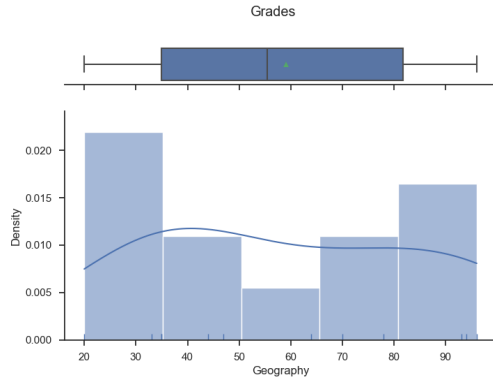
⁴<https://www.thoughtco.com/grading-on-a-curve-3212063?print>

⁵https://erikrood.com/Python_References/pandas_column_average_median_final.html

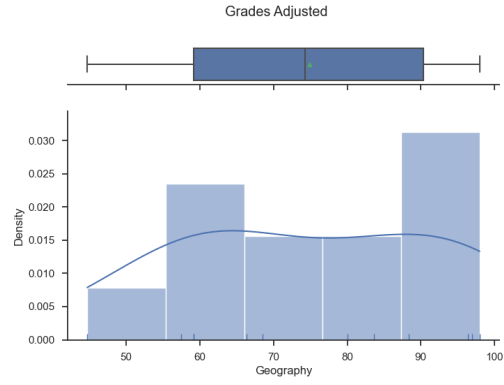
⁶<https://www.througheducation.com/how-does-grading-on-a-curve-work/>

⁷<https://www.thoughtco.com/grading-on-a-curve-3212063>

⁸<https://divisbyzero.com/2008/12/22/how-to-curve-an-exam-and-assign-grades/>

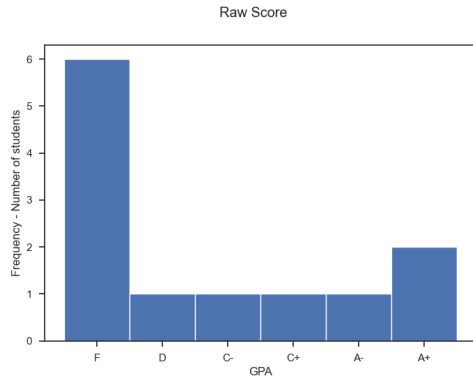


(a) Hist. original de Geo (média 5.9).

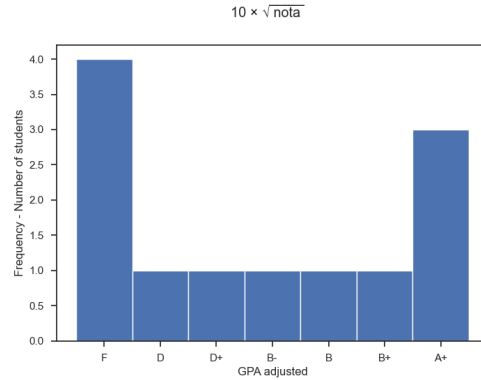


(b) Hist. ajustado de Geo (média 7.5).

Figura 3: Histogramas de frequência com os respectivos quartis.



(a) Hist. original de Geo (média 5.9).



(b) Hist. ajustado de Geo (média 7.5).

Figura 4: Histogramas de frequência dos GPAs.

qualquer, e assumir⁹ uma distribuição normal^{10 11 12 13}, dada pela equação 1 da figura 5, correspondendo às notas originais e às notas ajustadas, como pode ser visto nas figuras 2a e 2b.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp^{-\frac{1}{2} \times \left(\frac{x-\mu}{\sigma}\right)^2}, -\infty < x < \infty \quad (1)$$

onde:

- $f(x)$ = função de densidade de probabilidade.
- $P[x \leq nota] = \int_{-\infty}^{nota} f(x)dx$
- σ = desvio padrão.
- μ = média.

O eixo y de uma distribuição normal corresponde a uma densidade de probabilidade¹⁴, ou seja, a probabilidade de uma *nota* estar no intervalo $[a,b]$ é a área sob a curva neste intervalo. Como consequência, se $a = b$, a probabilidade de $nota = a$ é zero. No entanto, entender exatamente o que isso significa não é muito importante nesse trabalho.

⁹<https://www.geeksforgeeks.org/how-to-make-a-bell-curve-in-python/>

¹⁰<https://www.voitto.com.br/blog/artigo/distribuicao-normal>

¹¹<https://www.inf.ufsc.br/~andre.zibetti/probabilidade/normal.html>

¹²<http://www.est.ufmg.br/~edna/bionutri/NUT-Aula07.pdf>

¹³Também conhecida como uma curva em forma de sino - *bell curve*.

¹⁴https://pt.wikipedia.org/wiki/Fun%C3%A7%C3%A3o_de_densidade

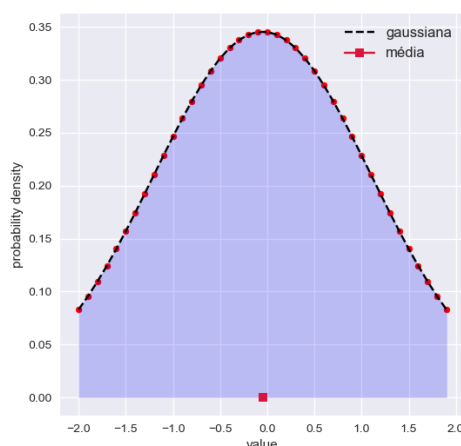


Figura 5: Distribuição Normal (Gaussiana).

Para que serve a distribuição normal nessa estória toda? Bem, ela serve apenas para termos uma interpretação/visualização dos dados de uma forma mais confortável. Embora tenhamos apenas 12 alunos na planilha, o normal seria haver 50, 60 ou até mesmo 100. Os gráficos servem então para rapidamente visualizarmos o desempenho da turma.

Permite também comparar a performance relativa em testes diferentes, ou seja, qual a proporção de pessoas que foram melhor ou pior que dois alunos, um que fez o teste A e outro que fez o teste B. Por exemplo, o aluno do teste A pode ter ido melhor do que 89.44% daqueles que fizeram o mesmo teste, enquanto o aluno que fez o teste B pode ter ido melhor do que 84.13% das pessoas que fizeram o teste B ^{15 16 17}. Comparando dessa forma, não faz muita diferença qual teste foi o mais fácil, ou quando os testes foram realizados (e.g., Enem de diferentes anos).

Nas figuras 3a e 3b podem ser vistos os histogramas de frequência¹⁸ e seus quartis¹⁹. Assim como as medianas dividem os dados em metades, de tal forma que 50% das notas caem abaixo da mediana e 50% caem acima, os quartis quebram os dados em quartos, de tal forma que 25% das notas são menores que o quartil inferior, 50% são menores que a mediana, e 75% são menores que o quartil superior. Note-se que as médias estão assinaladas com círculos verdes.

O programa deve permitir escolher, através da interface:

- pelo menos três curvas diferentes, incluindo a raiz quadrada, escala linear e nota mais alta vai para 100.
- a disciplina a ser ajustada.
- os gráficos a serem desenhados, com as curvas das notas (matplotlib)²⁰, e / ou os histogramas com seus quartis (seaborn)²¹.
- trocar as notas por conceitos para calcular o GPA (*Grade Point Average*)²², usando

¹⁵<https://towardsdatascience.com/how-to-use-and-create-a-z-table-standard-normal-table-240e21f36e53>

¹⁶https://www.w3schools.com/statistics/statistics_standard_normal_distribution.php

¹⁷<https://www.math.arizona.edu/~jwatkins/normal-table.pdf>

¹⁸<https://www.alura.com.br/artigos/o-que-e-um-histograma>

¹⁹<https://www.investopedia.com/terms/q/quartile.asp>

²⁰<https://stackabuse.com/matplotlib-line-plot-tutorial-and-examples/>

²¹<https://acervolima.com/boxplot-usando-seaborn-em-python/>

²²https://en.wikipedia.org/wiki/Academic_grading_in_the_United_States

os seguintes intervalos ²³:

- bins=[0, 60, 70, 80, 90, 100],
- labels=['F', 'D', 'C', 'B', 'A']

ou

- bins=[0, 60, 63.3, 66.7, 70, 73.3, 76.7, 80, 83.3, 83.7, 90, 93.3, 93.7, 100],
- labels=['F', 'D-', 'D', 'D+', 'C-', 'C', 'C+', 'B-', 'B', 'B+', 'A-', 'A', 'A+']

- salvar os conceitos numa nova planilha chamada *gpa.xlsx*

Código 2: Sistema americano de graus com letras

	Students	Class	History	Geography	Maths	Biology	Physics	Chemistry	Literature	Total
0	Student 1	B	F	F	D	F	F	C	F	340
1	Student 2	A	A	F	F	F	F	D	C	373
2	Student 3	A	D	F	F	F	F	F	F	279
3	Student 4	A	D	C	F	C	D	F	A	441
4	Student 5	C	F	A	F	A	D	F	F	422
5	Student 6	B	A	A	B	F	F	C	F	469
6	Student 7	C	B	C	D	F	F	C	F	405
7	Student 8	A	C	D	B	F	F	B	B	441
8	Student 9	B	F	F	D	F	C	F	F	1376
9	Student 10	C	F	F	A	F	C	F	F	337
10	Student 11	C	F	F	D	B	D	C	A	458
11	Student 12	A	B	A	F	A	A	B	B	542

	Students	Class	History	Geography	Maths	Biology	Physics	Chemistry	Literature	Total
0	Student 1	B	F	F	D	F	F	C+	F	340
1	Student 2	A	A-	F	F	F	F	D	C	373
2	Student 3	A	D-	F	F	F	F	F	F	279
3	Student 4	A	D	C-	F	C	D+	F	A-	441
4	Student 5	C	F	A-	F	A+	D	F	F	422
5	Student 6	B	A+	A+	B-	F	F	C+	F	469
6	Student 7	C	B-	C+	D	F	F	C+	F	405
7	Student 8	A	C	D	B+	F	F	B+	B-	441
8	Student 9	B	F	F	D	F	C+	F	F	1376
9	Student 10	C	F	F	A+	F	C-	F	F	337
10	Student 11	C	F	F	D-	B+	D	C-	A-	458
11	Student 12	A	B-	A+	F	A-	A-	B-	B-	542

Finalmente, deve ser entregue um código que funcione, ou seja, com todos arquivos .py, imagens, e qualquer outro recurso necessário à execução do programa.

Obs:

- Além do pandas²⁴, numpy²⁵, seaborn²⁶, e matplotlib²⁷, os únicos pacotes que podem ser instalados estão indicados no código1. Não serão aceitos programas idênticos ou que importem qualquer outro pacote externo.
- Serão aceitas apenas interfaces textuais baseadas no getopt²⁸, e.g.:

```
data.py --subject Literature --class=C --grade=87 --function sqrt
data.py -s Maths -c B -v
```

- Na interface textual, deve haver obrigatoriamente um opção `--help` ou `-h`. A opção `-v` controla a exibição dos gráficos. A opção `--grade` ou `-g` imprime apenas notas maiores do que o valor fornecido.

```
$ data.py -h
Usage ./data.py -s <subject> -c <class> -g <grade> -f<function> -v
Subjects = ['History', 'Geography', 'Maths', 'Biology', 'Physics', 'Chemistry', 'Literature']
```

Leiam as referências com cuidado, boa sorte e divirtam-se.

²³<https://pandas.pydata.org/docs/reference/api/pandas.cut.html>

²⁴<https://pandas.pydata.org>

²⁵<https://numpy.org>

²⁶<https://seaborn.pydata.org>

²⁷<https://matplotlib.org>

²⁸<https://docs.python.org/3/library/getopt.html>