

Programação Aplicada à Estatística

Pedro Rafael Diniz Marinho

Universidade Federal da Paraíba

2020.2

Sobre mim

Sou Pedro Rafael D. Marinho e serei o professor de vocês nesse período letivo. Sou Dr. em Estatística. Toda minha formação foi na área de Estatística (bacharelado, mestrado e doutorado).

Sou lotado no Centro de Ciências Exatas e da Natureza no Departamento de Estatística - UFPB.

A minha sala é a de número 12.

Meu email: `pedro.rafael.marinho@gmail.com`.

Trabalhei no mestrado com **modelos lineares com heteroscedasticidade de forma desconhecida** com orientação do PhD Francisco Cribari Neto. Na dissertação desenvolvemos simulações de bootstrap (simples e duplo) para avaliação da cobertura dos intervalos de confiança que indexam os parâmetros desses modelos. Um pacote na linguagem R foi desenvolvido.

Título da dissertação

Estimadores Intervalares Sob Heteroscedasticidade de Forma Desconhecida Via Bootstrap Duplo

O pacote `hcci` na versão 1.0.0 encontra-se hospedado no CRAN do R em <https://cran.r-project.org/web/packages/hcci/index.htm>.

No doutorado trabalhei com o PhD Gauss Moutinho Cordeiro na área de **distribuições de probabilidade**. Na tese foram criadas novas classes de distribuições de probabilidade em que é possível gerar uma nova distribuição a partir de uma distribuição G conhecida (baseline). Também foi construído o pacote `AdequacyModel` na linguagem R que encontra-se atualmente na versão 2.0.0 sob os termos da licença $GPL \geq 2$ (*GNU General Public License*)

Título da tese

Some New Families of Continuous Distributions

O pacote poderá ser obtido em <https://cran.r-project.org/web/packages/AdequacyModel/index.html>.

O curso de **Programação Aplicada à Estatística** é formado pela seguinte ementa.

Ementa do Curso

Introdução: Modelo de um computador digital; Linguagem de máquina; Introdução à Programação; Histórico das linguagens de programação; Compiladores e Interpretadores; Lógica e Lógica de Programação; Construção de algoritmos; Pseudocódigo.

Linguagem C: Visão geral; Expressões; Controle de fluxo; Funções; Ponteiros, Vetores e Matrizes; Alocação dinâmica de memória; Cadeias de caracteres; Tipos estruturados; Arquivos; Ordenação e Busca. Aplicações práticas à Estatística.

Plano de Curso

O **plano de curso** seguirá a ementa apresentada logo acima no *frame* anterior.

O plano de curso é de responsabilidade do professor da disciplina e deve estar de acordo com a ementa da disciplina.

O plano de curso é estabelecido pelo professor e ficará a cargo desse a sua construção e alteração, se necessário, no decorrer do curso, desde que o mesmo esteja de acordo com a ementa do curso.

O rigor e profundidade do assunto ficará a cargo do professor e faz parte do seu plano de curso.

Plano adotado no momento

1. Modelo de um computador digital. 2. Elementos básicos de um computador típico. 3. Armazenamento de dados e programas na memória. 4. Linguagem de máquina. 5. Histórico das linguagens de programação. 6. Tipos de linguagens. 7. Compiladores e Interpretadores. 8. Lógica e Lógica de programação 9. Construção de algoritmos. 10. Linguagem C. 11. Variáveis. 12. Operadores. 13. Entrada e Saída. 14. Controle de fluxo. 15. Construções de Laços. 16. Seleção. 17. Funções. 18. Ponteiros, Vetores e Matrizes. 19. Alocação dinâmica de memória. 20. Cadeia de caracteres. 21. Tipos estruturados. 22. Arquivos. 23. Ordenação e busca. 24. Aplicações práticas à Estatística.

Bibliografias Utilizadas

Abaixo estão enumeradas as bibliografias **básicas** utilizadas:

- ① Celes, W., Cerqueira, R., Rangel, J.L. Introdução a Estruturas de Dados: Com Técnicas de Programação em C. Rio de Janeiro: Elsevier, 2004.
- ② Kernighan, B. W., Ritchie, D. M. C: A linguagem de programação padrão ANSI. Rio de Janeiro: Elsevier, 1989.
- ③ Schildt, H. C Completo e Total. São Paulo: Makron Books, 1996.

Bibliografias Utilizadas

Abaixo estão enumeradas as bibliografias **básicas** utilizadas:

- ① Oliveira, U. Programando em C Volume I: Fundamentos. Rio de Janeiro: Ciência Moderna, 2008.
- ② Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P. Numerical recipes in C: the art of scientific computing, Cambridge : Cambridge University Press, 1994.

Recursos utilizados

Boa parte do curso será apoiada pelo uso de **datashow** o que nos ajudará bastante a decorrer sobre os diversos assuntos contidos na ementa desse curso que é bastante ampla.

O quadro será utilizado para resolução de alguns exemplos bem como complementações em que o professor achar conveniente no momento de aula.

Sobre as avaliações

No curso iremos considerar **três avaliações**, duas provas e **TALVEZ** um trabalho.

As duas primeiras avaliações serão provas que irão contemplar o conteúdo ministrado em sala de aula (**tudo que foi dito, apresentado e escrito**).

No caso em que for decidido por um trabalho como terceira avaliação, este deverá ser formado por um grupo de no máximo 3 pessoas.

Sobre as avaliações

As datas e temas dos trabalhos serão fornecidos após a segunda avaliação. A divisão dos grupos ficará a cargo dos alunos.

O professor tomará partido na divisão em caso de problemas.

Os trabalhos serão avaliados segundo sua organização, profundidade do assunto e irei avaliar o código que será fornecido no Apêndice do trabalho e enviado para meu email.

Importante

A reposição do trabalho será uma prova referente ao assunto do tema da equipe em que o aluno encontra-se presente.

Sobre as reposições

O aluno terá direito a apenas uma reposição de uma das avaliações desde que satisfeito o que rege a **Resolução N° 16/2015 que aprova o Regulamento dos Cursos Regulares de Graduação da UFPB.**

O aluno poderá repor uma prova desde que entre com pedido de reposição junto à coordenação do seu curso. O coordenador de seu curso irá avaliar o pedido de reposição com base na **Resolução N° 16/2015** do CONSEPE e encaminhará o pedido julgado ao Departamento de Estatística - UFPB.

Apenas irá repor a prova quem atender os requisitos do Art. 92, 6º §.

Aos alunos interessados (\LaTeX)

Aconselho fortemente ao aluno que pretende produzir textos de qualidade (qualidade tipográfica) considerar o uso da linguagem de comandos macros \LaTeX .

\LaTeX é uma linguagem de comandos macros de \TeX e atualmente encontra-se na versão $\text{\LaTeX} 2_{\epsilon}$.

Com o uso de \LaTeX a facilidade de construir texto de alta qualidade tipográfica será uma ferramenta a mais na mão de um profissional em estatística.

Com \LaTeX é possível fazer grandes mudanças em um texto em poucos minutos apenas acrescentando alguns comandos ao preâmbulo do código.

Aos alunos interessados (\LaTeX)



Figura: Donald Knuth.

\TeX é um sistema de tipografia científica desenvolvido por **Donald E. Knuth** que é orientado à produção de textos técnicos e fórmulas matemáticas. A pedido da AMS (*American Mathematical Society*), Donald Knuth desenvolveu uma linguagem de computador para editoração de textos com muitas equações.

Aos alunos interessados (\LaTeX)



Figura: Donald Knuth.

O trabalho de criação do \TeX se estendeu de 1977 a 1998, quando \TeX foi disponibilizado gratuitamente. O \TeX possui aproximadamente **600 comandos** que controlam a construção de uma página.

Pode-se considerar o \TeX como sendo um compilador para textos científicos que produz documentos de alta qualidade tipográfica.

Aos alunos interessados (\LaTeX)

O \TeX atingiu um estado de desenvolvimento em que Beebe (1990 afirmou):

“Meu trabalho no desenvolvimento de \TeX , METAFONT e as fontes Computer Modern chegou ao final. Eu não irei realizar mudanças futuras, exceto corrigir sérios erros de programação.”

Ver em: BEEBE, N. H. Comments on the future of \TeX and METAFONT. TUGboat, v. 11, n. 4, p. 490–494, 1990.

Aos alunos interessados (\LaTeX)

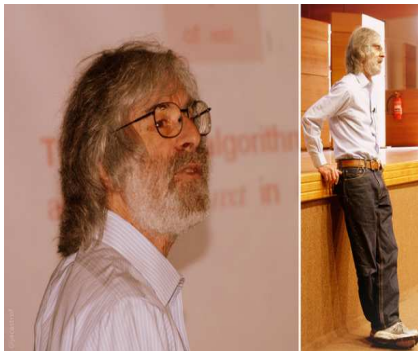


Figura: Leslie Lamport.

Quase que em paralelo foi desenvolvido por Leslie Lamport o \LaTeX . Essas macros definem tipos de documentos, tais como livros, artigos, cartas, entre outros.

Inclusive essa apresentação é um tipo básico de documento que foi produzido em \LaTeX .

Aos alunos interessados (\LaTeX)

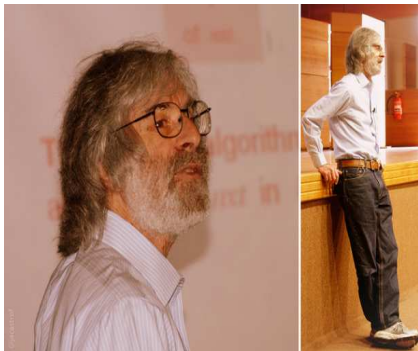


Figura: Leslie Lamport.

Para maiores detalhes leia sobre o pacote beamer que está disponível com a maioria das distribuições \LaTeX . Há diversos documentos disponíveis nos mais variados idiomas na rede.

beamer também está disponível no **The Comprehensive \TeX Archive Network (CTAN)**.

Aos alunos interessados (\LaTeX)



Figura: CTAN lion drawing
by Duane Bibby.

CTAN é o lugar central para todos os tipos de material em torno de \TeX . CTAN tem atualmente **5752 pacotes** e **2637 colaboradores** contribuíram para essa quantidade de pacotes.

O símbolo ao lado foi desenhado pelo artista comercial Duane Bibby. Este leão foi utilizado nas ilustrações para o livro \TeX Book de Donald Knuth e apareceu com grande frequência em outros materiais.

Aos alunos interessados (\LaTeX)



Figura: CTAN lion drawing
by Duane Bibby.

Maiores detalhes sobre o CTAN podem ser encontrados em <https://www.ctan.org/lion/>. Desde dezembro de 1994, a pacote \LaTeX está sendo atualizado pela equipe \LaTeX 3, dirigida por Frank Mittelbach, para incluir algumas melhorias que já vinham solicitadas a algum tempo. A equipe se preocupa também em reunificar todas as versões modificadas que surgiram desde o aparecimento do \LaTeX 2.09.

Aos alunos interessados (\LaTeX)



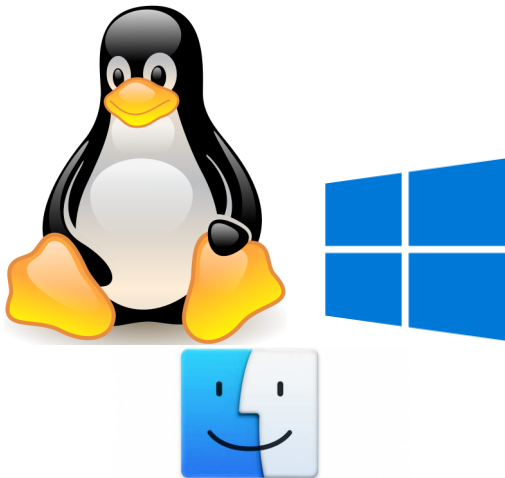
Figura: CTAN lion drawing
by Duane Bibby.

O melhor de tudo, o \LaTeX é um sistema estável mas com crescimento constante, podendo ser instalado em quase todos os sistemas operacionais.

Usuários de Unix, Linux, Windows ou Mac OS X podem dispor de todo ferramental para produzir ótimos textos com o \LaTeX .

Nota: Pronuncia-se “leitec” e não “latéx”.

Aos alunos interessados (\LaTeX)



MacTM OS

Como instalo o \LaTeX no Linux?

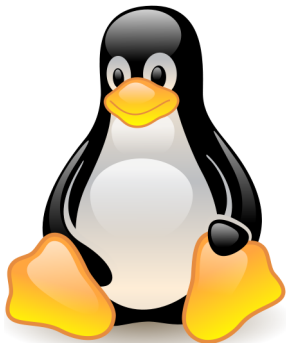


Figura: Tux (Mascote do Linux).

Como instalo o \LaTeX no Linux?

Inicialmente é preciso instalar o compilador de \LaTeX . Recomendo o uso do \TeX Live.

A maioria das distribuições linux (Arch, Ubuntu, Fedora, Mint, Sabayon, entre outros “sabores”) apresentam esse compilador de \LaTeX em seus repositórios.

Por exemplo, no **Arch Linux** e distribuições derivadas que utilizam os mesmos repositórios do Arch como Antergos façam:

```
sudo pacman -S texlive.
```

Como instalo o \LaTeX no Linux?

No **Ubuntu** ou qualquer distribuição que faz uso dos repositórios do Ubuntu façam:

```
sudo apt-get install texlive-full.
```

Já os usuários da distribuição **Fedora** e distribuições derivadas que utilizam-se dos mesmos repositórios devem fazer:

```
sudo dnf -y texlive-scheme-full.
```

Observação: Todos os comandos acima devem ser executados no terminal da respectiva distribuição com permissão de super usuário (usuário que pode fazer alterações no sistema operacional).

Como instalo o \LaTeX no Windows?



Como instalo o \LaTeX no Windows?

Felizmente, há o \TeX Live para Windows que poderá ser obtido no site oficial do projeto \TeX Live.

O usuário de Windows deverá baixar o arquivo

`install-tl-windows.exe`

que possui aproximadamente **13mb**.

Nota: `install-tl-windows.exe` é apenas o instalador do \TeX Live para Windows. Dessa forma, ao final da instalação, o \TeX Live terá muito mais que apenas 13mb instalado em seu computador.

Como instalo o \LaTeX no Linux?

Mas para escrevermos um texto com qualidade usando o \LaTeX precisamos também de um editor de texto.

Na maioria dos casos usamos uma **IDE** (*Integrated Development Environment*) (**Ambiente de Desenvolvimento Integrado**)

Aconselho o uso do \TeX studio que está disponível para Linux, Windows e Mac OS.

O \TeX studio é um software sobre os termos da licença **GPL** (*GNU General Public License*) e pode ser obtido em <http://texstudio.sourceforge.net/>.

Observação: Aperte F6 para compilar o documento e F7 para visualizar o PDF produzido por meio de código \LaTeX .

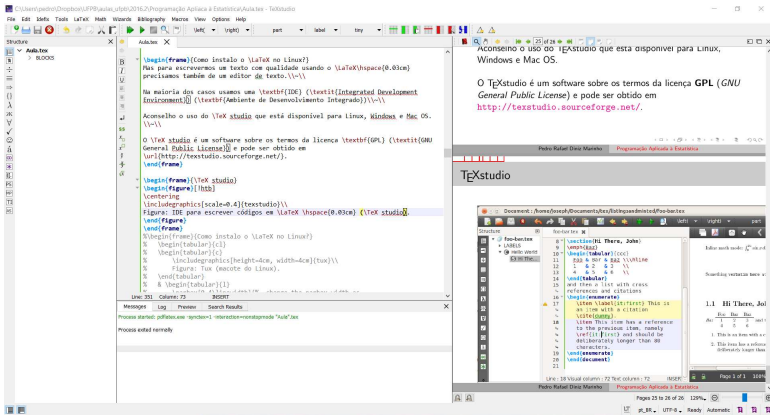


Figura: IDE para escrever códigos em \LaTeX (\TeX studio).

Vantagens do T_EXstudio

Vantagens do T_EXstudio

- 1 É uma IDE leve.

Vantagens do T_EXstudio

- ① É uma IDE leve.
- ② O T_EXstudio auto-completa os comandos que vão sendo digitados.

Vantagens do T_EXstudio

- ① É uma IDE leve.
- ② O T_EXstudio auto-completa os comandos que vão sendo digitados.
- ③ É possível ir de um ponto específico do código L^AT_EX para o ponto correspondente no PDF criado.

Vantagens do T_EXstudio

- ① É uma IDE leve.
- ② O T_EXstudio auto-completa os comandos que vão sendo digitados.
- ③ É possível ir de um ponto específico do código L^AT_EX para o ponto correspondente no PDF criado.
- ④ Tem licença **GPL** (*GNU General Public License*), isto é, não pago por ela e tenho acesso ao seu código fonte.

Vantagens do T_EXstudio

- ① É uma IDE leve.
- ② O T_EXstudio auto-completa os comandos que vão sendo digitados.
- ③ É possível ir de um ponto específico do código L^AT_EX para o ponto correspondente no PDF criado.
- ④ Tem licença **GPL** (*GNU General Public License*), isto é, não pago por ela e tenho acesso ao seu código fonte.
- ⑤ O PDF é visualizado ao lado do código L^AT_EX.

Como aprender \LaTeX ?

Pergunta do aluno

Okay professor, o senhor me convenceu em utilizar \LaTeX . Então, como eu posso aprender os seus comandos? Há algum material interessante para se começar a estudar \LaTeX ?

Resposta do professor legal

Há vários livros e materiais disponibilizados na internet. Há diversos grupos de discussão sobre \LaTeX . Para não complicar muito, aconselho estudar por essa apostila:

http://www.univasf.edu.br/~joseamerico.moura/index_arquivos/lenimar_tex_LATEX.pdf

O que é programação?

Resposta: Linguagem de programação é um método padronizado para comunicar instruções para um computador por meio de uma sintaxe.

Trata-se de um conjunto de regras sintáticas utilizadas para passar instruções para um computador. Por meio dessas regras, é possível que o programador especifique os **tipos de dados** em que o professor irá processar.

Tais dados serão armazenados e transmitidos e/ou transmitidos entre os componentes que forma o computador. Assim, a linguagem também permite especificar quais ações devem ser tomadas e sob quais circunstâncias serão tomadas.

Por que aprender à programar?

Por que um estatístico deve saber programar?

Por que aprender à programar?

Por que o estatístico deve saber programar?

Resposta: Simplesmente pelo fato de que **não** dá para fazer muita coisa na estatística (principalmente no mercado de trabalho) se o profissional não é capaz de fazer com que o computador resolva os seus problemas.

Um aluno questiona...

Mas professor, temos o SPSS, Excel, SAS, Statistica e outros softwares em que podemos chamar nosso conjunto de dados e apertar centenas de botões e ter **alguns** resultados.

Por que aprender à programar?

O professor continua respondendo...

A maioria desses software não possuem as técnicas estatísticas mais recentes e muitas vezes não são capazes de se adequar aos problemas específicos que nos deparamos ao tentar resolver um problema.

Muitas vezes precisamos modificar uma função programada por uma outro programador para que ela venha a funcionar no nosso problema.

Observação: Diversas outras vezes precisamos programar para realizar simulações. É muito comum na estatística estudar propriedades de algumas estatística ou modelo estatístico e querer simular o seu comportamento em diversos cenários diferentes.

Por que aprender à programar?

Uma linguagem de programação bastante utilizada na estatística é a linguagem R.

Linguagem R

R é uma linguagem de programação para computação estatística e gráficos. R é uma parte oficial do projeto GNU da Free Software Foundation's.

Curiosidade: A linguagem R foi criada originalmente por Ross Ihaka e Robert Gentleman no **Departamento de Estatística** da Universidade de Auckland, Nova Zelândia em agosto de 1993.

Nota: É muito importante que um estatístico saiba programar na linguagem R. Alguns empregos exigem isso. Porém, se não exigirem, o R te ajudará bastante.

Por que aprender à programar?



Figura: Criadores da linguagem R [Robert Gentleman (foto à esquerda) e Ross Ihaka (foto à direita)].

Por que aprender à programar?



Figura: Logo da linguagem R.

Um dos grandes motivos da grande popularidade da linguagem R se deve a grande quantidade de pacotes disponíveis para os usuários da linguagem.

Atualmente há mais de 5 mil pacotes para R com o foco nas mais variadas áreas: estatística, matemática, biologia, economia, entre outras.

Por que aprender à programar?



Figura: Logo da linguagem R.
Obtenha a linguagem R em
<https://www.r-project.org/>.

Observação: Para programar em R **não é suficiente entender alguns pacotes específicos**. É preciso entender a sintaxe base da linguagem que nos permite inclusive criar outros pacotes e melhorar os existentes.

Por que aprender à programar?



Figura: Logo da linguagem R.
Obtenha a linguagem R em

<https://www.r-project.org/>.

Observação: As novas metodologias estatísticas chegam mais rapidamente em R do que em outros softwares estatísticos pelo fato do R ser uma linguagem livre (código aberto e gratuita).

“Ciência da computação tem tanto a ver com o computador como a Astronomia com o telescópio, a Biologia com o microscópio, ou a Química com os tubos de ensaio. A Ciência não estuda ferramentas, mas o que fazemos e o que descobrimos com elas.” - **Edsger Dijkstra** (Prêmio Turing em 1972)



Figure: Edsger Dijkstra

Máquina de Turing

A máquina de Turing é um dispositivo teórico conhecido como máquina universal, que foi concebido pelo matemático britânico Alan Turing (1912-1954), muitos anos antes de existirem os modernos computadores digitais.

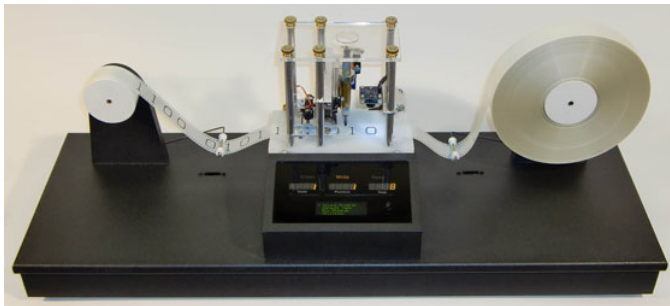


Figure: Exemplo de uma Turing física.