

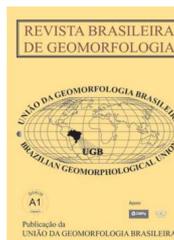


www.ugb.org.br
ISSN 2236-5664

Revista Brasileira de Geomorfologia

v. 18, nº 2 (2017)

<http://dx.doi.org/10.20502/rbg.v18i2.1158>



NOTA TÉCNICA:

ANÁLISE DIGITAL DE TERRENO UTILIZANDO A LINGUAGEM COMPUTACIONAL R: EXEMPLO DE APLICAÇÃO

DIGITAL TERRAIN ANALYSIS USING COMPUTATIONAL R LANGUAGE: EXAMPLE OF APPLICATION

Renê Jota Arruda de Macêdo

*Colegiado de Ciências da Natureza, Universidade Federal do Vale do São Francisco
Rua João Ferreira dos Santos, s/n –, São Raimundo Nonato, Piauí. CEP 64770-000. Brasil
Email: rene.macedo@univasf.edu.br*

Sergio Floquet

*Colegiado de Engenharia Civil, Universidade Federal do Vale do São Francisco
Avenida Antonio Carlos Magalhães, 510, Juazeiro, Bahia. CEP 48.902-300. Brasil
Email: sergio.floquet@univasf.edu.br*

Informações sobre o Artigo

Recebido (Received):
29/09/2016
Aceito (Accepted):
11/04/2017

Palavras-chave:

Linguagem R; Análise Digital de Terreno; Geomorfometria.

Keywords:

R Language; Digital Terrain Analysis; Geomorphometry.

Resumo:

Linguagens de programação vem se tornando populares em diversas áreas do meio acadêmico-científico. No âmbito das Geociências, emergem como potenciais ferramentas para a compreensão dos processos naturais da superfície terrestre. Neste trabalho realizou-se uma breve apresentação da linguagem computacional R e uma rápida abordagem a respeito da parametrização de elementos da superfície de uma determinada região a partir de dados discretos espaçados regularmente. Em seguida, apresenta-se um exemplo de sua aplicação para derivação de parâmetros geométricos e análise digital de terreno (ADT) em um modelo digital de elevação com 30 m de resolução espacial. Utilizou-se o *software* RStudio versão gratuita que oferece um ambiente de desenvolvimento gráfico intuitivo com diversas facilidades para implementação de rotinas. Com uma comunidade colaborativa ativa e aberta, a linguagem R aplicada em ADT permite que usuários iniciantes compreendam os aspectos básicos e visualize todo o processo de implementação do código e análise dos resultados.

Abstract:

Programming languages are becoming popular in many scientific fields. In the geosciences they provide potential tools for understanding natural processes on Earth's surface. In this paper we perform a brief introduction to the R programming language and a show a quick approach to the surface parameterization of a specific region in a regularized discrete data set. For example, to obtain geometric parameters, a digital terrain analysis (DTA) on a 30 m resolution digital elevation model was undertaken. The free version of *Rstudio* software used in this work

provides a graphical user interface environment with several facilities to implement routines. With an active and open collaborative community, R language applied in DTA allows beginner-level users to be able to understand basics aspects and visualize all process in the implementation and analysis of the results.

1. Introdução

Modelos Digitais de Elevação (MDE) consistem em representações numéricas da superfície da Terra. A partir deles é possível iterar técnicas para modelar processos hidrológicos além de extrair propriedades de solos, vegetação, insolação, processos sedimentares, dentre outros, salvaguardando o refinamento da escala do MDE (Wilson & Gallant, 2000).

A análise da morfologia de um terreno a partir de MDE vem sendo amplamente aplicada nas últimas décadas (Zevenbergen & Thorne, 1987; Grohmann, 2004; Jordan *et al.*, 2005, Grosse *et al.*, 2012), devido ao rápido processamento de dados por computadores, disponibilidade de imagens de sensor da superfície terrestre e o baixo custo nas análises. Os métodos em Análise Digital de Terreno (ADT) surgem do desenvolvimento de algoritmos para extração de uma ou múltiplas variáveis morfométricas do relevo a partir de uma imagem em formato matricial (*e.g.* Horn, 1981; Zevenbergen & Thorne, 1987; Ritter, 1987; Freeman, 1991). Através destes métodos é possível gerar Modelos Digitais de Terreno (MDT), que correspondem à funções discretas bidimensionais de variáveis morfométricas (Florinsky, 2012).

Nos últimos anos, o uso de linguagens de programação vem se destacando no meio acadêmico em áreas de conhecimento distintas daquelas em que são comumente utilizadas - engenharias, física, matemática e tecnologia da informação. Tal fato se deve ao surgimento de linguagens de alto nível, disponíveis gratuitamente, além de diversos textos escritos direcionados para leitores iniciantes. Dentre elas, a linguagem R, que, além do campo da estatística, vem sendo aplicada em estudos com dados espacialmente coordenados, tais como morfológicos e geoestatísticos (Grohmann, 2004; Metcalfe *et al.*, 2015; Kobal *et al.*, 2013; Omuto & Vargas, 2014; Brown, 2016).

Neste artigo a linguagem computacional R será utilizada para derivação de dados morfométricos e ADT. Serão introduzidos os aspectos básicos da linguagem, assim como a sua aplicação em um MDE, apresentando os conceitos fundamentais da geomorfometria numa abordagem parametrizada do relevo. Todos os métodos

executados estão explicitados com scripts comentados a fim de que qualquer leitor possa realizar suas próprias ADT e modificar conforme sua necessidade. Além de divulgar a linguagem R, de distribuição livre, o presente trabalho pretende tornar mais acessível as implementações básicas, tendo como foco profissionais e pesquisadores das geociências e áreas afins.

2. A Linguagem Computacional R

R foi originalmente desenvolvido para análise de dados estatísticos e gráficos (Ihaka & Gentleman, 1996), com sintaxe e semântica semelhantes a linguagem comercial S. Atualmente, vem ganhando popularidade nos diversos ramos acadêmicos além da estatística. Nas geociências, sua potencialidade é brevemente descrita por Grunsky (2002), além de outros exemplos como nos trabalhos de Bivand (2000), que realizou análises geoestatísticas, e Grohmann (2004), o qual extraiu parâmetros morfométricos do relevo através de um MDE utilizando o *software Geographic Resources Analysis Support System* (GRASS GIS), tratando os dados estatisticamente com a linguagem R dentro do ambiente GRASS.

O sistema R é constituído de uma interface gráfica mais a linguagem em si, distribuído livremente. Há abundante material instrutivo disponível em seu próprio repositório chamado de CRAN (*Comprehensive R Archive Network*) no endereço <https://cran.r-project.org/>, além de outras diversas fontes como sites e livros. O ambiente de desenvolvimento possui várias funções estatísticas embutidas que podem ser executadas diretamente na linha de comando de sua interface gráfica, além de compatibilidade com outras linguagens de programação, tais como: C, C++, Fortran, Java e Python.

2.1 Programação e Ambiente de Desenvolvimento R

O sistema R possui seu próprio ambiente de desenvolvimento, a RGui, formado por uma interface gráfica composta de facilidades para o desenvolvedor onde se dá a entrada, armazenamento, manipulação e visualização gráfica de dados (R Core Team, 2014). A execução dos comandos pode ser realizada diretamente

no console, ao digitar e em seguida apertando a tecla ENTER, ou no script, semelhante a um bloco de notas, sendo através da seleção da linha de código que deseja executar e apertando as teclas CTRL + R.

Para iniciantes que não tenham noções básicas sobre conceitos de programação ou da sintaxe e semântica do R, a interface RGui pode restringir o avanço no desenvolvimento de seus primeiros códigos, devido à ausência de funções de autocorreção e auto completar que auxiliam e agilizam a implementação dos códigos. Existem outros ambientes de desenvolvimento, sendo um dos mais populares o RStudio (RStudio Team, 2015). Seu ambiente possui uma interface amigável, dividida em 4 janelas (Figura 1): 1) canto superior esquerdo, a janela de script (caso não exista nenhuma aberta, basta

clique no menu file → new file → R Script); 2) canto superior direito, janela de área de trabalho (*workspace*); 3) canto inferior esquerdo, janela do console e; 4) canto inferior direito, janela de arquivos, plots, pacotes, ajuda e visualizador. O acesso a ajuda no RStudio também pode ser realizado executando a função `help()`, com o nome do pacote ou da função dentro dos parênteses, desde que este pacote tenha sido previamente carregado. Uma janela com a documentação é apresentada com todos os detalhes – descrição, uso, argumentos, referências e exemplos – do comando desejado. Maiores informações sobre funções, bibliotecas, importação/exportação de dados, operadores, exemplos e exercícios, podem ser encontradas no documento introdutório de Torfs & Brauer (2014).

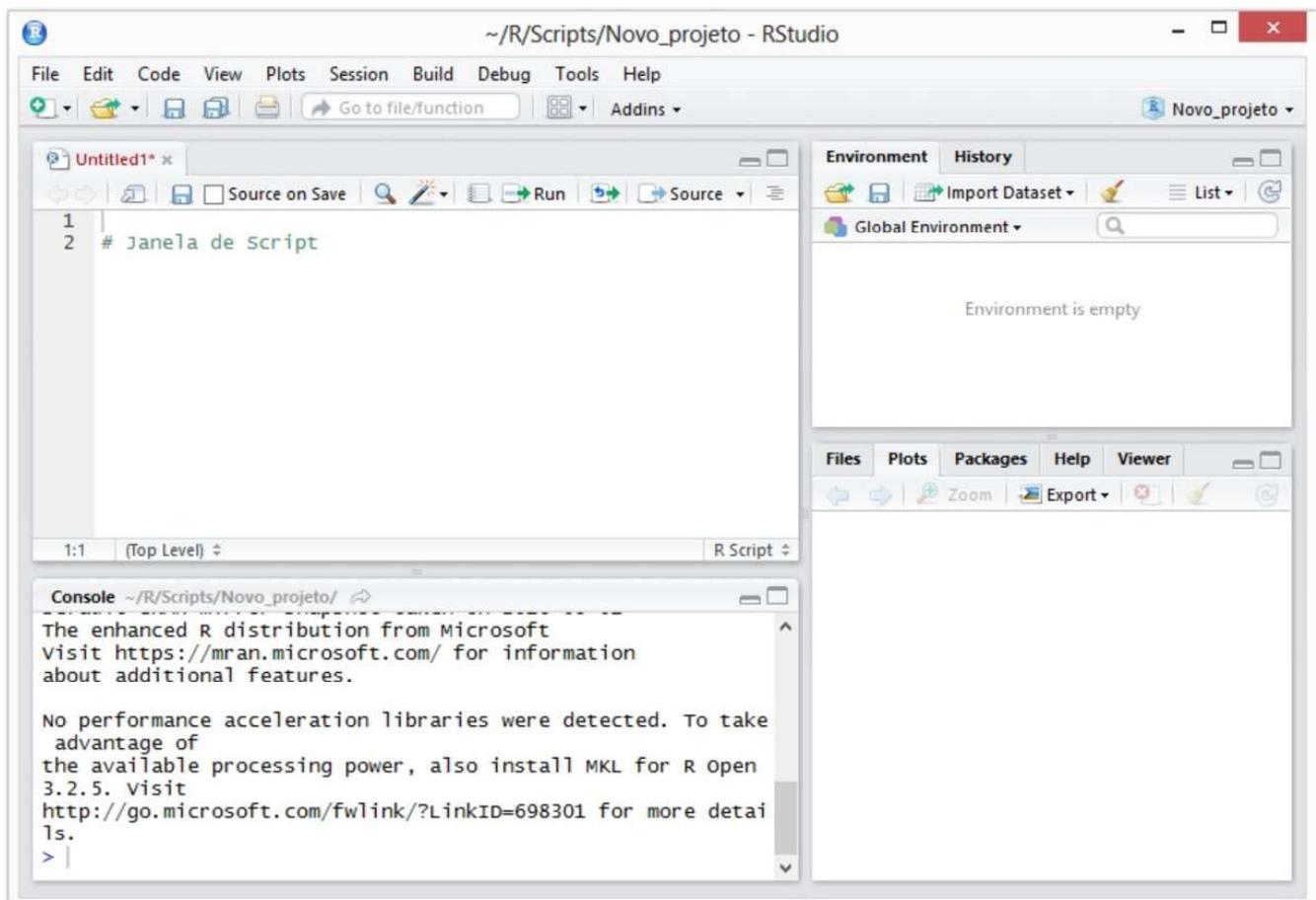


Figura 1 - Interface gráfica do software RStudio.

Antes de iniciar qualquer projeto, é importante que o usuário determine o diretório de trabalho, para que os arquivos gerados sejam guardados na pasta previamente definida. No menu *session* → *set working directory* → *choose directory*, escolha o local desejado para o ambiente de trabalho do R.

2.2 Manipulação de Dados Tipo Raster

Na linguagem R podemos manipular diferentes tipos de dados, tais como: vetores, matrizes, fatores, conjunto de dados, listas, rasters e outros, importando e exportando arquivos de diversos formatos (Zuur *et*

al., 2009). MDE's são imagens de dados tipo raster e consistem em conjuntos numéricos dispostos em grades, cujas células são espaçadas regularmente, representando a superfície topográfica de uma determinada área da Terra. Cada célula corresponde a um valor de elevação, coordenado sob um sistema de referência geoespacial.

No R, um raster é definido utilizando um pacote adicional, que permite a criação e manipulação deste objeto. A instalação e ativação de pacotes podem ser realizadas diretamente pelo console do RStudio, através dos seguintes comandos:

```
> install.package("raster") # função que instala o pacote
> library(raster) # ativa o pacote para uso
```

O download da versão mais atual do pacote, após o primeiro comando, é executado direto da CRAN. Ao completar a instalação, o pacote poderá ser ativado pelo comando `library()` ou `require()` e a imagem importada pela função `raster()`. A fim de facilitar a importação, a imagem deverá estar na mesma pasta definida como diretório de trabalho.

importando a imagem

```
> raster_x <- raster("08S435ZN.tif")
> raster_x # retorna os atributos do objeto
```

Em alguns casos, pode ocorrer incompatibilidade, no momento da instalação, entre o pacote e a versão do RStudio. Nesta situação, o melhor será obter as versões mais recentes do RStudio ou utilizar versões mais antigas do pacote raster que sejam compatíveis, este último, pode ser obtido diretamente na CRAN em *Old sources* (Fontes antigas – em tradução literal) do pacote. O pacote raster, demanda de outros pacotes que dão suporte às operações de suas funções. Estes são chamados de dependências e, no geral, são instalados automaticamente junto com o pacote raster. Não obstante, em raras exceções, as dependências podem não ser instaladas. Caso isso ocorra, o usuário deverá instalá-las individualmente com a função `install.packages('nome_da_dependência')`. As dependências podem ser descobertas quando na tentativa

de utilizar alguma função do pacote raster ocorrer um erro indicando a ausência destas ou através da leitura do manual do pacote, disponível no site do repositório de pacotes (CRAN).

As operações básicas são executadas com raster de maneira usual, em que a soma, subtração, multiplicação, divisão e potenciação são feitas elemento a elemento (*i.e.* por célula do raster), assim como a aplicação das diversas funções tais como seno, cosseno, tangente, logaritmo, etc. Se quisermos exagerar os valores de elevação em 10x, basta escrever:

```
> raster_10x <- raster_x*10
```

Os principais parâmetros geomorfológicos de um MDE retratam suas características locais, que são determinadas a partir da relação de um ponto com seus vizinhos. Para realizar operações a fim de obter atributos morfométricos do relevo, é necessário definir uma função em termos das coordenadas *x* e *y* que irá representar a superfície local e daí extrair os parâmetros desejados (Hengl & Reuter, 2009). Em um raster (Figura 2), toma-se uma janela formada por uma matriz 3x3, constituída de 8 elementos vizinhos mais o ponto central em análise. A superfície definida pela matriz de ordem 3 é aproximada por uma função polinomial. Calcula-se os parâmetros desejados a partir desses 9 elementos por meio da aplicação de um método de diferenças finitas e o valor resultante é atribuído ao elemento central. O processo repete-se em toda a extensão da imagem, gerando um novo objeto com os atributos computados.

A função `focal()`, do pacote raster, permite computar parâmetros de um MDE definindo-se o tamanho de uma grade regular de *n* vizinhos, retornando o resultado na célula central desejada. Como exemplo no script a seguir, aplicou-se a função `focal()` ao raster dados, definindo-se uma grade de análise 3x3, para retornar o maior valor desta grade. Para contornar o problema das bordas do MDE, assumiu-se aqui o valor 1 para o argumento `padValue` da função, a qual cria vizinhos fictícios para os elementos das bordas:

```
> calc.dados <- focal(raster_x,
w=matrix(1,nrow=3, ncol=3), fun=max,
pad=TRUE, padValue=1)
```

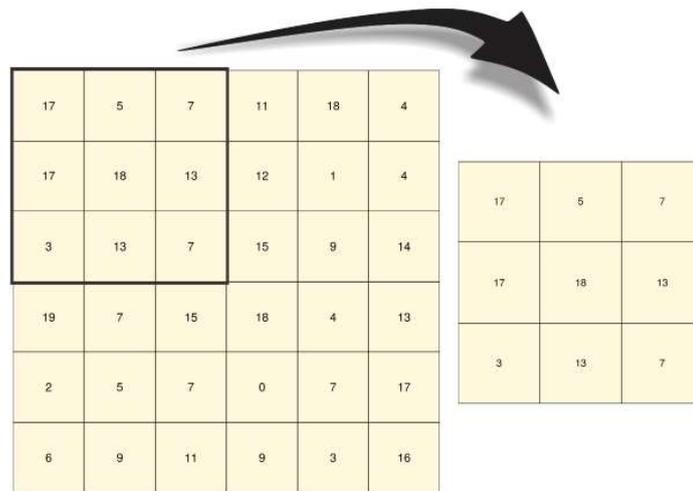


Figura 2 - Esquema de um raster onde destaca-se uma grade regular 3 x 3 para computação dos atributos locais.

3. Análise Digital de Terreno

Parte do estudo dos parâmetros de um MDE consiste na análise de aspectos geométricos da superfície do terreno. Para efetuar esta abordagem matemática é necessário termos uma função bivariada $z = f(x, y)$, contínua para derivações de até segunda ordem, representando a superfície da grade analisada e que os parâmetros topográficos possam ser expressos por derivadas desta função (Mitasova & Hofierka, 1993).

Existem diversas funções na literatura utilizadas para representar a superfície de um MDE. Evans e Young (Young & Evans, 1978; Pennock *et al.*, 1987) aplicaram uma função quadrática para os 9 elementos de uma submatriz 3x3 para derivar 4 índices topográficos (Equação 1):

$$z(x,y) = \frac{rx^2}{2} + sxy + \frac{ty^2}{2} + px + qy + z_0 \quad (1)$$

As constantes p, q, r, s e t da função $z(x,y)$ estão relacionadas às derivadas parciais, enquanto z_0 representa o valor da função na origem. A partir da derivação da função no ponto (x,y) na origem $(0,0)$, que corresponde ao centro da célula da submatriz, obtém-se as constantes:

$$\frac{\partial Z}{\partial x} \Big|_{(0,0)} = (rx + sy + p) \Big|_{(0,0)} = p \quad (2)$$

$$\frac{\partial Z}{\partial y} \Big|_{(0,0)} = (ty + sx + q) \Big|_{(0,0)} = q \quad (3)$$

$$\frac{\partial^2 Z}{\partial x^2} \Big|_{(0,0)} = r \quad (4)$$

$$\frac{\partial^2 Z}{\partial x \partial y} \Big|_{(0,0)} = s \quad (5)$$

$$\frac{\partial^2 Z}{\partial y^2} \Big|_{(0,0)} = t \quad (6)$$

Ao determinar os valores das constantes p, q, r, s e t encontram-se as derivadas parciais de primeira e segunda ordem da função $z(x,y)$. Então, substituiu-se os 9 valores da submatriz 3x3 (Figura 3) na Equação 1, gerando um sistema de equações lineares.

Z1 = Z(-1,1)	Z2 = Z(0,1)	Z3 = Z(1,1)
Z4 = Z(-1,0)	Z5 = Z(0,0)	Z6 = Z(1,0)
Z7 = Z(-1,-1)	Z8 = Z(0,-1)	Z9 = Z(1,-1)

Figura 3 - Coordenadas da vizinhança de um raster.

Desta forma encontra-se um conjunto de 9 equações envolvendo as altitudes de cada célula da submatriz 3x3 {Z1, Z2, ..., Z9}, expressas por:

$$Z_1 = \frac{r}{2} - s + \frac{t}{2} - p + q + z_0 \quad (7)$$

$$Z_2 = \frac{t}{2} + q + z_0 \quad (8)$$

$$Z_3 = \frac{r}{2} + s + \frac{t}{2} + p + q + z_0 \quad (9)$$

$$Z_4 = \frac{r}{2} - p + z_0 \quad (10)$$

$$Z_5 = z_0 \quad (11)$$

$$Z_6 = \frac{r}{2} + p + z_0 \quad (12)$$

$$Z_7 = \frac{r}{2} + s + \frac{t}{2} - p - q + z_0 \quad (13)$$

$$Z_8 = \frac{t}{2} - q + z_0 \quad (14)$$

$$Z_9 = \frac{r}{2} - s + \frac{t}{2} + p - q + z_0 \quad (15)$$

A solução desse sistema de equações lineares pode ser obtida utilizando a representação matricial $[Z] = [A] * [Constantes]$, em que $[Z]$ é a matriz 9x1 com os valores de elevação, $[A]$ a matriz 9x6 com os coeficientes que surgem multiplicando as constantes da função $z(x,y)$ e a matriz 6x1 $[Constantes]$, formada pelas constantes p, q, r, s, t e z_0 que irá ser determinada (Equação 16).

$$\begin{pmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \\ Z_7 \\ Z_8 \\ Z_9 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & -1 & \frac{1}{2} & -1 & 1 & 1 \\ 0 & 0 & \frac{1}{2} & 0 & 1 & 1 \\ \frac{1}{2} & 1 & \frac{1}{2} & 1 & 1 & 1 \\ \frac{1}{2} & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{1}{2} & 0 & 0 & 1 & 0 & 1 \\ \frac{1}{2} & 1 & \frac{1}{2} & -1 & -1 & 1 \\ 0 & 0 & \frac{1}{2} & 0 & -1 & 1 \\ \frac{1}{2} & -1 & \frac{1}{2} & 1 & -1 & 1 \end{pmatrix} \begin{pmatrix} r \\ s \\ t \\ p \\ q \\ z_0 \end{pmatrix} \quad (16)$$

As derivadas parciais que representam os coeficientes da função, que passam a ser definidas em função dos valores de elevação da submatriz. Tais constantes, podem ser determinadas solucionando o sistema da Equação 16 isolando a matriz $[Constantes]$. Para isso, é necessário transformar $[A]$ numa matriz quadrada, multiplicando pela sua transposta $[A]^T$ e em seguida, calcular a sua inversa (Equação 17).

$$[A]^T [Z] = [A]^T [A] [Constantes] \\ [Constantes] = \{ [A]^T [A] \}^{-1} [A]^T [Z] \quad (17)$$

Essa operação pode ser realizada utilizando o R com o comando `t()` para obter a transposta da matriz, a função `solve()` calcula sua inversa e o símbolo `%` é utilizado para indicar produto entre matrizes:

```
> solve(t(A)% * % A)%%t(A)
```

Assim os parâmetros $\{p, q, r, s, t, z_0\}$, que correspondem as derivadas parciais de primeira e segunda ordem da função $z(x,y)$ do modelo de Evans-Young (Young & Evans, 1978), são:

$$p = \frac{\partial Z}{\partial x} = \frac{Z_3 + Z_6 + Z_9 - Z_1 - Z_4 - Z_7}{6 \cdot \Delta s} \quad (18)$$

$$q = \frac{\partial Z}{\partial y} = \frac{Z_1 + Z_2 + Z_3 - Z_7 - Z_8 - Z_9}{6 \cdot \Delta s} \quad (19)$$

$$r = \frac{\partial^2 Z}{\partial x^2} = \frac{Z_1 + Z_3 + Z_4 + Z_6 + Z_7 + Z_9 - 2 \cdot (Z_2 + Z_5 + Z_8)}{3 \cdot \Delta s^2} \quad (20)$$

$$s = \frac{\partial^2 Z}{\partial x \partial y} = \frac{-Z_1 + Z_3 + Z_7 - Z_9}{4 \cdot \Delta s^2} \quad (21)$$

$$t = \frac{\partial^2 Z}{\partial y^2} = \frac{Z_1 + Z_2 + Z_3 + Z_7 + Z_8 + Z_9 - 2 \cdot (Z_4 + Z_5 + Z_6)}{3 \cdot \Delta s^2} \quad (22)$$

$$z_0 = \frac{5 \cdot Z_5 + 2 \cdot (Z_2 + Z_4 + Z_6 + Z_8) - (Z_1 + Z_3 + Z_7 + Z_9)}{9} \quad (23)$$

em que Δs é o tamanho da resolução espacial da célula do MDE.

Definidas as derivações é possível obter os atributos topográficos básicos locais resultantes das derivadas parciais de primeira ordem: gradiente e aspecto; e de segunda ordem: perfil de curvatura, plano de curvatura e curvatura tangencial. Os parâmetros locais são aqueles

computados para uma célula central de uma subgrade onde as operações dependem diretamente dos elementos vizinhos (Hengl & Reuter, 2009). Parâmetros regionais são aqueles em que consideram-se todo o conjunto de células de um MDE para se computar determinado atributo (e.g. bacias hidrográficas, linhas de fluxo, fluxo acumulado, sombreamento do relevo, etc.). Em geral, estes parâmetros dependem dos atributos locais para serem extraídos. Na definição de linhas de fluxos sobre uma determinada área, utilizam-se os valores do gradiente e seu ângulo azimutal (aspecto), que influenciam diretamente nos processos de escoamento de um fluido (Quinn *et al.*, 1991; Freeman 1991).

Em ADT, pode-se realizar uma descrição e análise isolada de cada variável computada ou de forma combi-

nada, a fim de identificar a distribuição e variabilidade de processos superficiais específicos (Moore *et al.*, 1991). Deste modo, a ADT possibilita a integração de vários campos de pesquisa relacionando as características topográficas a processos naturais. São exemplos de alguns ramos de aplicação: a influência da topografia na formação de solos, identificação de estruturas geológicas, caracterização de processos de movimentos de massa e estudos de bacias hidrográficas (Florinsky, 2012; Moore *et al.*, 1991; Jordan *et al.*, 2005).

Na Tabela 1, encontram-se resumidamente os atributos topográficos básicos locais e regionais com respectivas formulações, escala de aplicação e os processos geomórficos que podem ser descritos através destes.

Tabela 1: Relação de atributos básicos e secundários em análise de MDE. Modificado de Horn (1981), Zevenbergen & Thorne (1987), Freeman (1991), Mitasova & Hofierka, (1993), Wilson & Gallant (2000) e Hengl & Reuter (2009).

Atributo	Escala de Aplicação	Descrição	Equação	Processos Associados
Elevação	Local/Regional	Cotas altimétricas.	-	Clima, vegetação, energia potencial.
Declividade	Local	Gradiente de elevação.	$S = \arctan(\sqrt{p^2 + q^2})$	Taxa de fluxo aquoso e sedimentar.
Aspecto	Local	Sentido do gradiente de elevação.	$A = 180 - \arctan(\frac{q}{p}) + 90.(\frac{p}{ p })$	Escoamento primário de fluxo.
Perfil de Curvatura	Local	Taxa de variação da declividade.	$Kp = \frac{rp^2 + 2spq + tq^2}{(p^2 + q^2)(1 + p^2 + q^2)^{3/2}}$	Agradação e Erosão. Aceleração e desaceleração de uma fluido.
Plano de Curvatura	Local	Curvatura transversa à direção da declividade. (Curvatura de contorno)	$Kh = \frac{rq^2 - 2spq + tp^2}{(p^2 + q^2)^{3/2}}$	Convergência e divergência de um fluido.
Curvatura Tangencial	Local	Curvatura transversa e perpendicular à direção da declividade. (Curvatura de contorno)	$\frac{\partial^2 Z}{\partial x \partial y} \Big _{(0,0)} = s$	
Fluxo Acumulado	Regional	Escoamento superficial.	$A_i = \frac{Max(0, S_i^p)}{\sum_{j=i}^8 (Max(0, S_j^p))}$	Volume de <i>runnof</i> .
Relevo Sombreado	Regional	Mapa de reflectância e sombra do relevo.	$R = \cos(S). \cos(\theta_0) + \sin(S). \sin(\theta_0). \cos(\phi - A)$	Irradiação Solar; Morfotectônica (Estruturas geológicas).
Rugosidade	Local/Regional	Complexidade do terreno (grau de ondulação)	$R = \sqrt{1 + \tan(\alpha)^2}$	Morfoestruturas; Distribuição e riqueza de espécies.

θ_0 = ângulo zenital solar

ϕ = ângulo azimutal solar

4. Materiais e Métodos

A execução deste trabalho desenvolveu-se a partir dos seguintes passos: 1) obtenção do ambiente de desenvolvimento RStudio; 2) implementação das bibliotecas básicas para ADT; 3) aquisição da imagem em formato raster de uma área de interesse; 4) pré-processamento da imagem e; 5) computação dos parâmetros do terreno.

4.1 Ambiente de Desenvolvimento e Pacotes

A ADT realizada neste trabalho foi executada utilizando o *software* RStudio Desktop, versão aberta. Consiste numa interface de desenvolvimento integrado (*Integrated Development Environment - IDE*) que inclui ferramentas de edição e suporte à linguagem R. O *software* pode ser baixado no site do seu desenvolvedor (www.rstudio.com) disponível para os sistemas operacionais Windows, Mac ou Linux.

Apesar do RStudio possuir várias bibliotecas estatísticas embutidas, foi necessário a instalação de pacotes desenvolvidos especificamente para a computação de parâmetros morfométricos a partir de imagens rasters. Os pacotes estão disponíveis no repositório da comunidade de desenvolvimento da linguagem R (*The Comprehensive R Archive Network (CRAN)*) no endereço eletrônico <https://cran.r-project.org/>. O procedimento de instalação dos pacotes desejados se faz através da execução do script diretamente no console:

```
> install.packages('nome_do_pacote')
```

É importante certificar-se do acesso à internet, pois o programa instalará automaticamente, a partir do repositório, a biblioteca requerida e as dependências (*i.e.* os pacotes que dão suporte à biblioteca). Os pacotes utilizados na ADT dispõem-se na Tabela 2 com suas respectivas funções.

Tabela 2: Pacotes utilizados e suas respectivas funções em ADT.

Pacote	Descrição	Referência
raster	Leitura, escrita, manipulação e análise de dados espaciais regulares (matriz).	Robert J. Hijmans (2015). raster: Geographic Data Analysis and Modeling. R package version 2.5-2. https://CRAN.R-project.org/package=raster
rgdal	Operações com sistemas de projeções e transformações de projeções da biblioteca PROJ4.	Roger Bivand, Tim Keitt and Barry Rowlingson (2016). rgdal: Bindings for the Geospatial Data Abstraction Library. R package version 1.1-8. https://CRAN.R-project.org/package=rgdal
dynatopmodel	Modelagem de parâmetros hidrológicos do terreno. Análise de dados hidrológicos de séries históricas.	Peter Metcalfe, Keith Beven, Jim Freer (2016). dynatopmodel: Implementation of the Dynamic TOPMODEL Hydrological Model. R package version 1.1. https://CRAN.R-project.org/package=dynatopmodel
prettymapr	Pacote gráfico para adição de símbolos (escala, seta de norte, etc.) em mapas.	Dewey Dunnington (2016). prettymapr: Scale Bar, North Arrow, and Pretty Margins in R. R package version 0.1.5. https://CRAN.R-project.org/package=prettymapr

4.2 Modelo Digital de Elevação (MDE)

O MDE utilizado no presente artigo foi adquirido gratuitamente no site do Instituto Nacional de Pesquisas Espaciais (INPE) do projeto TOPODATA, que dispõe de imagens altimétricas no formato Geotiff, da

superfície do relevo brasileiro adquiridas pela missão *Shuttle Radar Topography Mission (SRTM)* da NASA. No endereço eletrônico institucional (www.dsr.inpe.br/topodata) é possível fazer o download a partir das quadriculas codificadas com as coordenadas geográficas

utilizando recursos do Googlemaps. Os exemplos aqui demonstrados foram executados com a imagem da quadrícula 8S435, correspondente a região do Parque Nacional Serra da Capivara, no sul do estado do Piauí. A imagem adquirida contém apenas valores de elevação, com resolução espacial aproximada de 1 arco-segundo, que corresponde a cerca de 30 m.

4.3 Parâmetros Morfométricos do Relevo

Propriedades geométricas extraídas da análise da superfície de um terreno são conhecidas como parâmetros morfométricos (Hengl & Reuter, 2009). Os atributos considerados como primários ou básicos da análise de terreno são aqueles que podem ser computados diretamente dos dados topográficos e secundariamente pela composição dos atributos primários. Os parâmetros computados neste trabalho foram os mesmos mencionados previamente e seguem na ordem de aparição da Tabela 1.

5. Exemplo de Aplicação

Aqui serão exemplificados procedimentos básicos com a linguagem R para extração de parâmetros geomorfológicos do relevo. Os índices morfométricos básicos como declividade, aspecto, curvatura tangencial (plano de curvatura), perfil de curvatura e rugosidade são derivados a partir de Modelos Digitais de Elevação (Hengl & Reuter, 2009).

5.1 Pré-processamento

Antes de iniciar qualquer procedimento com as imagens, deve-se carregar as bibliotecas que serão utilizadas para o pré-processamento através dos comandos a seguir:

```
> library(raster)
> library(rgdal)
> library(dynatopmodel)
```

As operações exigem que a projeção do raster esteja em coordenadas métricas. Para verificar a projeção execute o comando:

```
> proj4string(raster_x) # retorna o valor em string
```

A imagem utilizada no exemplo possui apenas as coordenadas geográficas (lat/long) sem *datum* de referência. O INPE informa que as imagens SRTM estão referenciadas no *datum* WGS84. É necessário realizar uma inserção do *datum* WGS84 para uma posterior conversão do mesmo para o datum SAD69 zona 23S que corresponde a área em que se encontra a quadrícula da região do PNSC em medidas métricas. Os códigos de *datum* de referência podem ser consultados no site <http://spatialreference.org/> que contém um acervo completo da biblioteca proj4.

Os procedimentos para a reprojeção da imagem a ser trabalhada estão descritos nos script abaixo com os comentários descrevendo o código:

```
# atribuir o datum de referência - crs(x)
<- 'valor'
> crs(raster_x) <- '+proj=longlat +datum=WGS84
+no_defs +ellps=WGS84 +towgs84=0,0,0'

# verificar a projeção
> crs(raster_x)

# define nova projeção
> novaproj <- '+proj=utm +zone=23 +south
+ellps=aust_SA +units=m +no_defs'

# raster reprojetado
raster_reprojetado <- projectRaster(raster_x,
crs=CRS(novaproj))

# verifica o resultado
> plot(raster_reprojetado)
```

O processamento do raster reprojetado poderá levar alguns minutos, dependendo da configuração do computador em que esteja sendo executado o *software* R. Ao término do processo, é necessário ajustar a resolução do novo raster para que os valores de x e y sejam iguais. Na imagem original, os valores da resolução, após a reprojeção, foram de 30,7 m. Neste caso, ajustamos a resolução para 30 m, que corresponde a aproximadamente 1 arco-segundo em coordenadas geográficas, conforme apresentado no fluxograma de refinamento do raster no site do INPE (INPE, 2016). Antes de ajustar a resolução, cria-se uma cópia do raster reprojetado para reamostrá-lo posteriormente.

```
# cópia do raster reprojetoado  
> raster_reprojetoado_aux <- raster_reprojetoado  
  
# mudando a resolução do raster reprojetoado  
> res(raster_reprojetoado) <- 30 # altera a  
resolução de x e y para 30m
```

Após definir a resolução, o raster reprojetoado perderá seus valores altimétricos, sendo necessário reamostrar os valores originais para a nova imagem. Aqui usa-se a cópia gerada antes do ajuste da resolução para ser reamostrado.

```
# reamostra os valores dos pixels do raster  
inicial para o novo raster  
# o método de reamostragem pode ser o 'biline-  
ar' ou 'ngb' = vizinhos mais próximos  
> raster_reprojetoado <- resample(raster_  
reprojetoado_aux, raster_reprojetoado,  
method='bilinear')
```

```
> raster_reprojetoado # confere os valores do  
novo raster
```

5.2 Processamento de Parâmetros Morfométricos

Após a fase de pré-processamento, podemos agora utilizar o raster de elevação reprojetoado na resolução e datum apropriados para derivação dos parâmetros morfométricos do relevo.

5.2.1 Elevação

A elevação corresponde a valores de cotas altimétricas do terreno em relação ao nível do mar em que cada ponto está referenciado espacialmente por um sistema de coordenadas. O mapa de elevação é adquirido rapidamente sem qualquer cálculo sobre os pontos da imagem, dado que o próprio raster consiste nos dados coletados de altitude de uma região. Assim, um MDE pode ser construído por meio da leitura direta do arquivo sem considerar os valores de dados vizinhos ou qualquer algoritmo, exceto quando é requerido uma interpolação, quando um ponto de interesse não coincide com o grid construído (Deng, 2007). Para tal, faz-se apenas a plotagem dos dados com uma classificação baseada em cores para gerar uma escala de classes de altitude (Figura 4).

Com base na imagem pré-processada (`raster_reprojetoado`) aplica-se o seguinte script:

```
# plota o gráfico  
> plot(raster_reprojetoado, maxpixels = 2000000,  
interpolate = TRUE, col=rev(rainbow(255)), xlim =  
c(664709.8,830909.8), ylim = c(9003651,9115551),  
legend.args=list(text='Elevação (m)', side=4,  
font=2, line=2.5, cex=0.8))
```

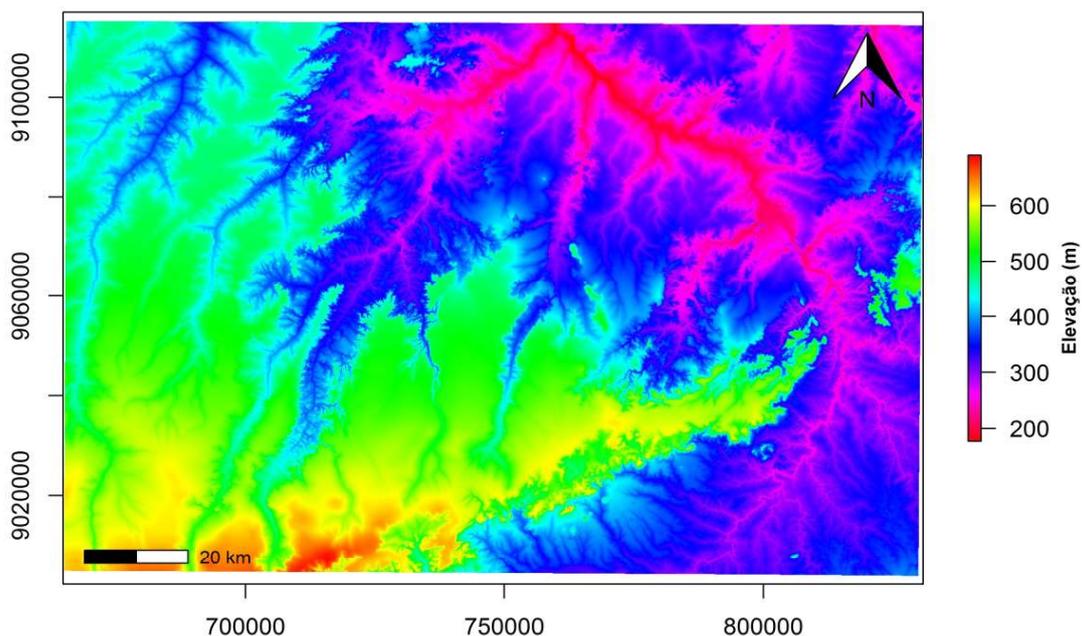


Figura 4 - Mapa de elevação da região do PNSC e entorno.

```
# adiciona escala  
> addscalebar(ploteps = 29193, widthhint  
= 0.15, style = 'bar', pos = 'bottomleft')
```

```
# adiciona seta do norte  
> addnortharrow(pos = 'topright', scale  
= 0.8)
```

5.2.2 Declividade

A declividade (S) é a taxa de variação da elevação na direção do declive. A derivação desse parâmetro é computada escolhendo-se a quantidade de células vizinhas. O pacote raster utiliza dois métodos baseados na quantidade de células vizinhas à célula central, podendo ser 4 (Ritter, 1987) ou 8 (Horn, 1981) numa grade de 3x3. Executou-se o procedimento para ambos os modelos, de modo que

os dados resultantes geraram imagens com diferenças na ordem de 0,01, coerente com as análises realizadas por Jones (1998). Para o exemplo, preferiu-se manter o padrão de células com 8 vizinhos conforme será utilizado nos demais parâmetros (Figura 5).

```
# cálculo da declividade  
> declividade <- terrain(raster_reprojetado,  
opt=c('slope'), unit='degrees', neighbors=8)  
# plota o gráfico  
> plot(declividade, maxpixels = 2e6, interpolate  
= TRUE, col=gray.colors(50, start=0.0, end=1.0,  
gamma=2.2), xlim = c(664709.8,830909.8),  
ylim = c(9003651,9115551), legend.  
args=list(text='Declividade (°)', side=4,  
font=2, line=2.5, cex=0.8))
```

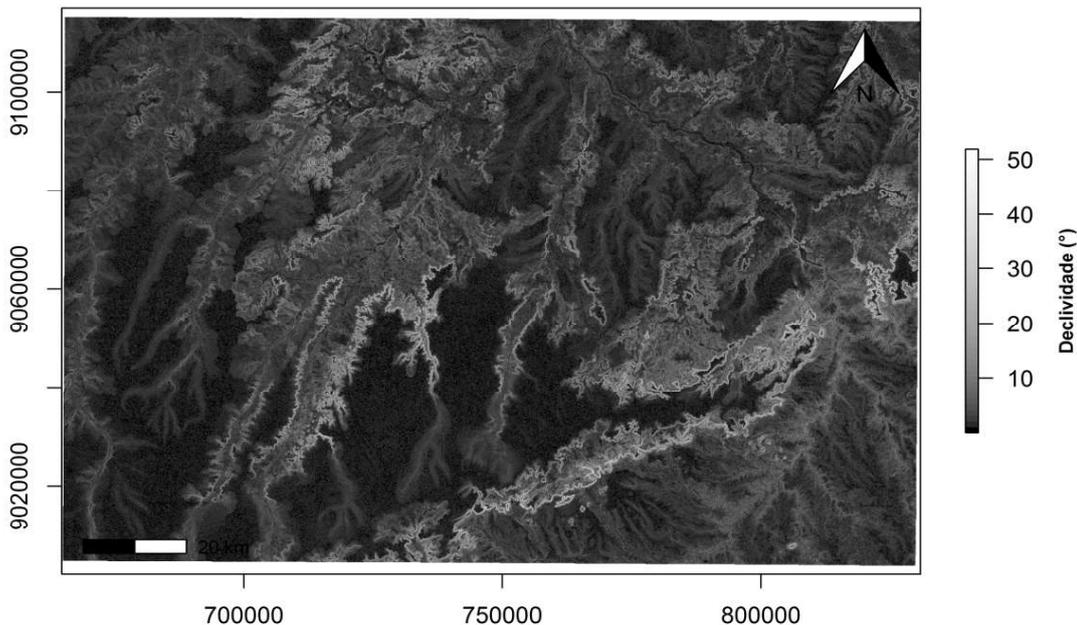


Figura 5 - Mapa de declividade.

5.2.3 Aspecto

O aspecto é o azimute da declividade, indicando a direção de movimento de um determinado fluido (Zevenbergen & Thorne, 1987). É comumente utilizado para visualizar e estimar radiação solar (Wilson & Gallant, 2000) e computar linhas de fluxo utilizadas em estudos hidrológicos (Hengel & Reuter, 2009). A direção da declividade (aspecto) é computada da mesma forma que a declividade, utilizando a biblioteca raster, através da função terrain:

```
# cálculo do aspecto  
> aspecto <- terrain(raster_reprojetado,  
opt=c('aspect'), unit='degrees', neighbors=8)
```

No script a seguir, optou-se por intervalar as direções do aspecto em classes incrementadas de 45°, pois um gradiente de cores dificultaria a visualização da tendência de sentido de caimento. O resultado do mapa pode ser visualizado na Figura 6.

```
# cria uma paleta de cores com a função color-
RampPalette
> my.colors <- colorRampPalette(c('red', 'yel-
low', 'green', 'cyan', 'blue', 'magenta'))

# gera intervalos de valores nos dados do raster
brks <- c(0.1e-05, 45.0, 90.0, 135.0, 180.0,
225.0, 270.0, 315.0, 360.0)

# cria intervalos de valores na legenda e indica
os valores do azimute
```

```
> arg <- list(at=c(0.1e-05, 45, 90, 135, 180,
225, 270, 315, 360), labels=c(0, 45, 90, 135,
180, 225, 270, 315, 360))
```

```
# plota o gráfico
```

```
> plot(aspecto, maxpixels = 2e6, interpolate =
TRUE, col=my.colors(8), axis.args=arg, xlim =
c(664709.8,830909.8), ylim = c(9003651,9115551),
legend.args=list(text='Azimute', side=4, font=2,
line=2.5, cex=0.8))
```

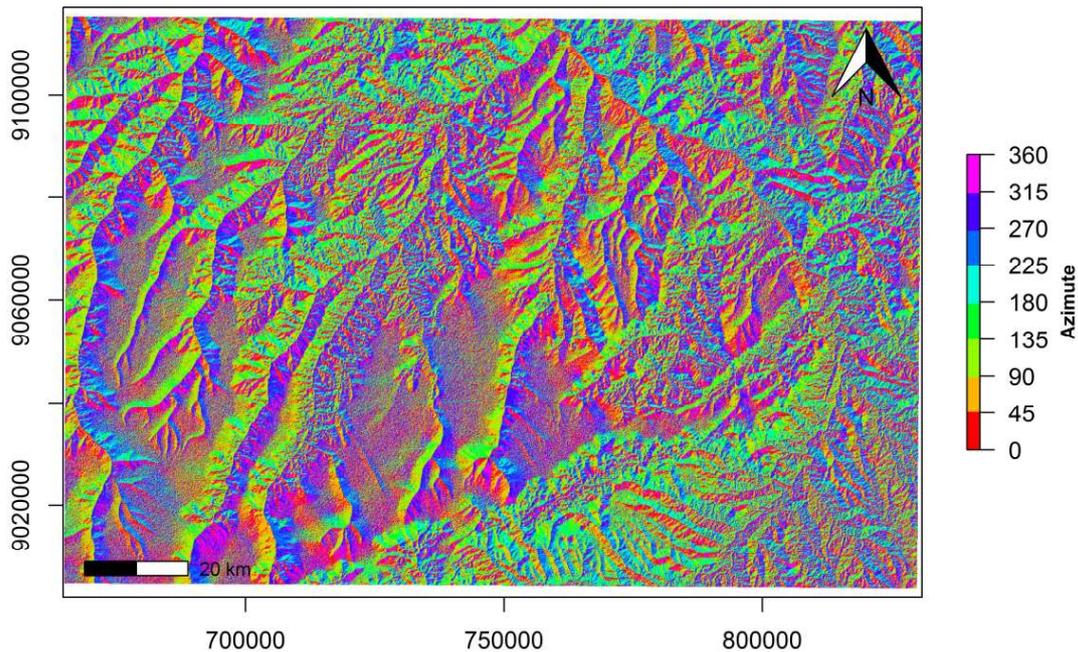


Figura 6 - Mapa do azimute das declividades.

5.2.4 Relevo Sombreado

Imagens sombreadas do relevo são produzidas utilizando os mapas de declividade e aspecto do terreno sobre os quais incidem uma única fonte de luz, o Sol, cuja posição é determinada pelo seu ângulo zenital e azimutal. O mapa resultante é uma imagem em escala de cinza na qual as áreas mais escuras não ocorrem devido à incidência direta da luz devido à existência de barreiras topográficas. Este tipo de mapa sombreado realça feições morfológicas (e.g. vales, planaltos e redes de drenagem) permitindo uma visualização rápida das formas superficiais. Uma das aplicações é a possibilidade de modelar a posição do Sol na esfera celestial e computar a quantidade de irradiação solar na superfície terrestre (Florinsky 2012), *i.e.* os efeitos da topografia no clima

de uma região (Böhner & Antonic, 2009). Deste modo, é possível simular o mundo real com a posição do Sol num particular horário e estação do ano (Zhou, 1992).

Mapas de relevo sombreado são fundamentais em análises morfoestruturais, pois aumentam o contraste de variações sutis de uma imagem facilitando a interpretação de feições geomorfológicas lineares associadas às fraturas (Jordan *et al.*, 2005). Esta técnica é comumente mais efetiva do que a interpretação de imagens de sensores remotos em termos de reconhecimento de falhas crustais (Florinsky, 2011). Ainda, segundo Florinsky (2011), uma vez identificadas por meio da abordagem geomorfométrica, o uso de mapas de dados geológicos torna-se fundamentais no auxílio da classificação das morfoestruturas. A topografia, em geral, expressa os

falhamentos presentes e podem ser associados com formações lineares assimétricas nas quais um dos lados apresentam quebras abruptas do gradiente ou escarpas, padrões lineares com comprimentos de ordem de dezenas a centenas de quilômetros (Dhont & Chorowicz, 2006).

Na computação do relevo sombreado para a área de exemplo, fez-se um pequeno procedimento *a priori* com a imagem de elevação. Por corresponder a uma região de terrenos predominantemente sedimentares muito planos, exagerou-se os valores da altitude a fim de que fosse possível realçar feições mais sutis de áreas menos acidentadas de modo que maiores elevações produzam mais zonas de sombras. O procedimento é bastante simples, considerando a quantidade de vezes que se deseja ampliar, basta multiplicar a imagem raster de elevação pelo valor requerido. O passo seguinte é produzir imagens rasters de declividade e aspecto (repetir os passos das suas respectivas seções para esses parâmetros) a partir da elevação exagerada para que o mapa de relevo sombreado produza os efeitos visuais esperados. A função utilizada para computar o relevo sombreado no R foi a `hillshade()` do pacote `raster`. Esta função tem como entrada as imagens de declividade e aspecto para a produção do relevo sombreado. Além disso, na função `terrain()`, devem ser definidos os ângulos zenital e azimutal para simular a posição da radiação solar sobre o relevo.

O mapa resultante pode ser visualizado na Figura 7 e no script abaixo encontram-se os procedimentos comentados:

```
# aumenta em 10x o valor da elevação
> raster_reprojetado_10x <- raster_reprojetado*10

# gerar mapas de declividade e aspecto
> slope_10x <- terrain(raster_reprojetado_10x,
  opt='slope', unit='radians', neighbors=8)
> aspecto_10x <- terrain(raster_reprojetado_10x,
  opt='aspect', unit='radians', neighbors=8)

# computar relevo sombreado
> sombreado <- hillshade(slope_10x,
  aspecto_10x, angle = 40, direction = 270)

# plota o gráfico resultante
> plot(sombreado, col=grey.colors(100,
  start=0, end=1, gamma = 0.45, alpha = NULL),
  legend= FALSE, xlim = c(664709.8,830909.8),
  ylim = c(9003651,9115551), maxpixels =
  2e6,interpolate=T)
```

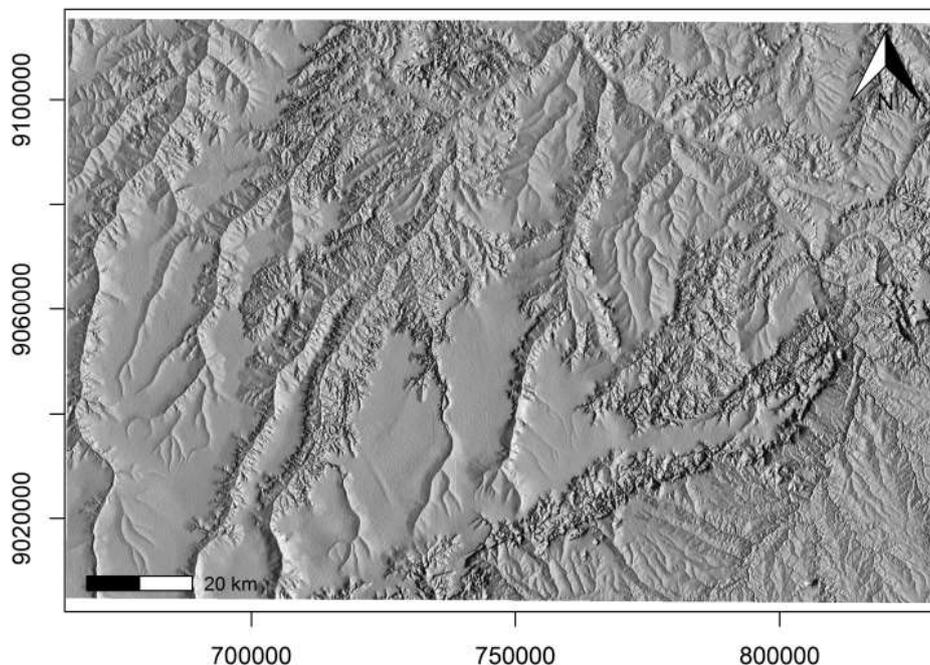


Figura 7 - Mapa de relevo sombreado com exagero vertical de 10 vezes.

5.2.5 Curvaturas

Em ADT extrai-se comumente 3 parâmetros de curvatura: (1) Perfil de curvatura, (2) Plano de Curvatura e (3) Curvatura tangencial. Numa superfície curva, cuja orientação da declividade é definida, o **perfil de curvatura** (K_p) é obtido a partir da interseção de um plano vertical à linha de orientação da declividade. Numa superfície curva, cuja orientação da declividade é definida para um ponto específico, o **plano de curvatura** (K_h) é a taxa de variação ortogonal à curva, também conhecida como linha de contorno de uma curvatura (Hengl & Reuter, 2009). A **curvatura tangencial** (K_t) deriva da curvatura da seção do plano normal à orientação da declividade e perpendicular ao gradiente (Mitasova & Hofierka, 1993). Em outras palavras, K_t é a curvatura que passa na interseção entre a declividade e um plano inclinado. De acordo com Mitasova & Hofierka (1993), a curvatura tangencial, por ser perpendicular ao gradiente, é mais apropriada para descrever a convergência/divergência do fluxo que o plano de curvatura. Ainda, é possível expressar K_t como o produto entre o seno da declividade e o plano de curvatura (Equação 24):

$$K_t = \text{sen}S \cdot K_h \quad (24)$$

Verifica-se que as curvaturas K_t e K_h possuem isolinhas de zero iguais, em que as áreas côncavas e convexas são as mesmas, mas divergem nos valores já que $K_h = K_t / \text{sen}S$ (Mitasova & Hofierka, 1993). A Figura 8 dispõe de um esquema das linhas de curvatura para uma determinada superfície num ponto específico. Os perfis de curvatura e curvatura tangencial podem ser combinados para definir a geometria básica das formas de relevo (Mitasova & Hofierka, 1993). A formulação destes parâmetros decorre da segunda derivação da função polinomial. Os valores resultantes serão positivos e negativos em radianos por metro e indicam as formas côncavas ou convexas no sentido do gradiente. Valores negativos, na curvatura tangencial, denotam convexidade da superfície, favorecendo a convergência de linhas de fluxo. Os positivos indicam concavidade da superfície levando a divergência de linhas de fluxo. Por exemplo, o plano de curvatura, ou curvatura de contorno, com valores negativos indica que a superfície é convexa com fluxos divergentes sobre uma crista, ao passo que com valores positivos há uma convexidade em que os fluxos convergem num vale (Wilson & Gallant, 2000).

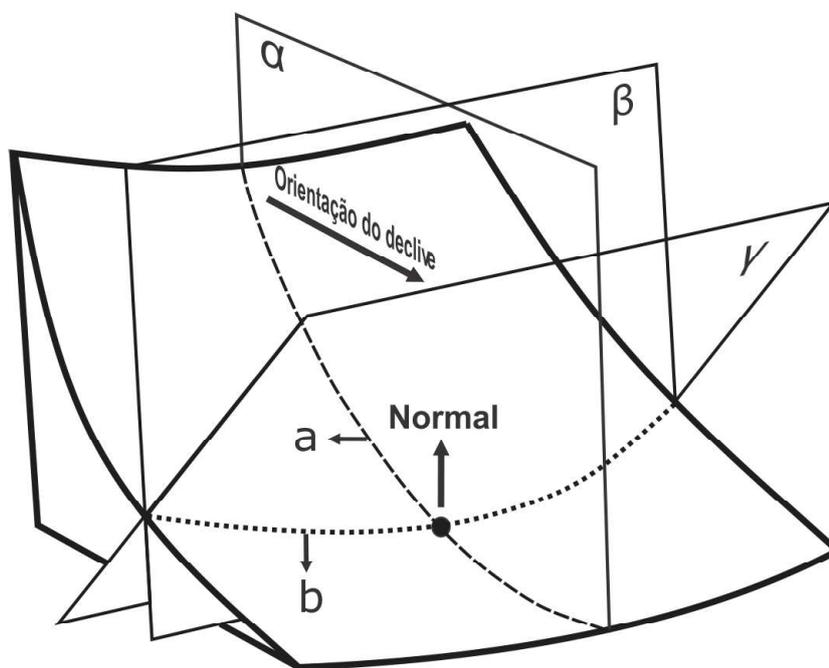


Figura 8 - Esquema de linhas de curvatura em gradiente côncavo. Os planos α , β e γ intersectam-se no ponto. (a) corresponde a linha do perfil de curvatura; (b) é a linha dos planos de curvatura e curvatura tangencial que sobrepõem-se pelo intersecto entre os planos β e γ . O plano γ é perpendicular ao sentido da linha do gradiente.

Os resultados podem ser utilizados para identificar relações de acúmulo de água no desenvolvimento de solos, modelar processos erosivos e inferir relações fitogeomorfológicas (interdependência de plantas com atributos do relevo – Moore *et al.*, 1991).

No R, computou-se as curvaturas a partir da definição da função quadrática de 6 termos polinomiais baseada no método Evans-Young para uma matriz 3x3, conforme Equação 1. A formulação permite enquadrar os 9 elementos de uma matriz de modo a extrair as segundas derivações encontradas para as constantes $r = \frac{\partial^2 Z}{\partial x^2}$, $s = \frac{\partial^2 Z}{\partial x \partial y}$ e $t = \frac{\partial^2 Z}{\partial y^2}$ necessárias para o cálculo das curvaturas. Baseando-se na expressão bidimensional de uma curvatura no plano (Equação 25), Young & Evans (1978) definiram as formulações para as curvaturas no plano e em perfil (Zevenbergen & Thorne, 1987).

$$K = \frac{\frac{\partial^2 Z}{\partial S^2}}{\left\{1 + \left(\frac{\partial Z}{\partial S}\right)^2\right\}^{3/2}} \quad (25)$$

A partir das equações das curvaturas disponíveis na Tabela 1, implementou-se três funções para calculá-las. No início de cada função, atribui-se o valor da resolução sendo a distância entre cada célula. Em seguida, para um determinado *raster* *x* de entrada, definiu-se a posição de cada célula para uma submatriz de análise 3x3. Conhecendo-se quais células compõem cada constante, as mesmas são declaradas e utilizadas ao final na fórmula da curvatura desejada.

```
# função perfil de curvatura
perfil <- function(x){
delta <- res(raster_reprojetado)[1] # resolução
z1<-x[1]; z2<-x[2]; z3<-x[3]; z4<-x[4]; z5<-x[5]; z6<-x[6]; z7<-x[7]; z8<-x[8]; z9<-x[9]
p <- (z3+z6+z9-z1-z4-z7)/(6*delta)
q <- (z1+z2+z3-z7-z8-z9)/(6*delta)
r <- (z1+z3+z4+z6+z7+z9-2*(z2+z5+z8))/(3*(delta)^2)
s <- (-z1+z3+z7-z9)/(4*(delta)^2)
t <- (z1+z2+z3+z7+z8+z9-2*(z4+z5+z6))/(3*(delta)^2)
resultado <- (r*(p^2)+2*p*q*s+t*(q^2))/(((p^2)+(q^2))*(sqrt((1+(q^2)+(p^2))^3)))
c(resultado)}
```

```
# função plano de curvatura
plano <- function(x){
delta <- res(raster_reprojetado)[1]
z1<-x[1]; z2<-x[2]; z3<-x[3]; z4<-x[4]; z5<-x[5]; z6<-x[6]; z7<-x[7]; z8<-x[8]; z9<-x[9]
p <- (z3+z6+z9-z1-z4-z7)/(6*delta)
q <- (z1+z2+z3-z7-z8-z9)/(6*delta)
r <- (z1+z3+z4+z6+z7+z9-2*(z2+z5+z8))/(3*(delta)^2)
s <- (-z1+z3+z7-z9)/(4*(delta)^2)
t <- (z1+z2+z3+z7+z8+z9-2*(z4+z5+z6))/(3*(delta)^2)
resultado <- (r*(q^2)-2*p*q*s+t*(p^2))/(sqrt((1+(p^2)+(q^2))^3))
c(resultado)}
```

```
# função curvatura tangencial
tangencial <- function(x){
delta <- res(raster_reprojetado)[1]
z1<-x[1]; z2<-x[2]; z3<-x[3]; z4<-x[4]; z5<-x[5]; z6<-x[6]; z7<-x[7]; z8<-x[8]; z9<-x[9]
p <- (z3+z6+z9-z1-z4-z7)/(6*delta)
q <- (z1+z2+z3-z7-z8-z9)/(6*delta)
r <- (z1+z3+z4+z6+z7+z9-2*(z2+z5+z8))/(3*(delta)^2)
s <- (-z1+z3+z7-z9)/(4*(delta)^2)
t <- (z1+z2+z3+z7+z8+z9-2*(z4+z5+z6))/(3*(delta)^2)
resultado <- (r*(q^2)-2*p*q*s+t*(p^2))/(((p^2)+(q^2))*(sqrt(1+(p^2)+(q^2))))
c(resultado)}
```

Após a definição das funções para as curvaturas, utilizou-se a função `focal()`, embutida no pacote `raster`, que permite aplicar a função criada para cada submatriz e realiza o movimento da quadrícula 3x3 por toda imagem raster. Para que a `focal()` realize os procedimentos necessários, dá-se como seus argumentos de entrada o raster utilizado (`srtm_reprojetado`), o tamanho da quadrícula, a função da curvatura, ativação da variável que suaviza os efeitos de borda adicionando-se células virtuais (`pad=TRUE`) e o valor destas células (`padValue=344`), sendo aqui utilizado o valor médio das células da imagem.

```
# Perfil de Curvatura
perfil_curvatura <- focal(srtm_reprojetado,w=m
```

```

atrix(1,nrow=3,ncol = 3),fun=perfil, pad=TRUE,
padValue=344)
# plano de curvatura
plano_curvatura <- focal(srtm_reprojetado,w=m
atrix(1,nrow=3,ncol= 3),fun=plano, pad=TRUE,
padValue=344)
# curvatura tangencial
tangencial_curvatura <- focal(srtm_reprojeta
do,w=matrix(1,nrow=3,ncol= 3),fun=tangencial,
pad=TRUE, padValue=344)

```

Os mapas de curvaturas gerados (Figura 9) são muito semelhantes visualmente e difíceis de serem individualizados na escala trabalhada. A distribuição dos valores resultantes é fortemente simétrica e próxima a zero, necessitando de uma paleta de cores que permita a diferenciação das curvaturas com limites de gradientes em torno de zero. O procedimento realizado para gerar os limites de cores para uma determinada paleta de cores está exemplificado logo a seguir para o mapa de perfil de curvatura. Os passos repetem-se para os demais atributos, diferindo apenas dos limites mínimos e máximos, dada a característica individual de cada curvatura calculada.

As quebras (*breaks*) indicam em quais valores deverão ocorrer as tonalidades escolhidas. Após alguns testes, os intervalos que melhor permitiram uma individualização das cores foram de 0.000 até ± 0.0025 . Os valores mais negativos e positivos destas escalas são determinados a partir do máximo e mínimo encontrado para a imagem.

```

# cria uma nova paleta de cores
> my.colors <- colorRampPalette(c('blue','cya
n','yellow','red3'))

# cria intervalos de quebra para as cores
> brks <- c(-0.01606651, -0.0015, -0.001,
-0.0005, -0.00025, -0.00015, 0.0, 0.00015,
0.00025, 0.0005, 0.001, 0.0025, 0.02291046)

# indica a posição dos valores na escala de cor
> arg <- list(at=c(-0.01606651, -0.008033255,
0.0, 0.01145523, 0.02291046), labels
=c(-0.016,-0.008,0.0,0.011,0.022))

# plota o gráfico
> plot(curvatura_perf, maxpixels = 2e6, inter
polate = TRUE,
col=my.colors(13),
breaks=brks, axis.args=arg,
xlim = c(664709.8,830909.8), ylim =
c(9003651,9115551))

```

Através dos histogramas de cada mapa (Figura 9) observa-se o grau de assimetria destes dados de curvaturas e corroboram com a afirmação de Mitasova & Hofierka (1993) onde as curvaturas do plano e tangencial divergem em valores, sendo a última (Figura 9c) correspondente do perfil de curvatura.

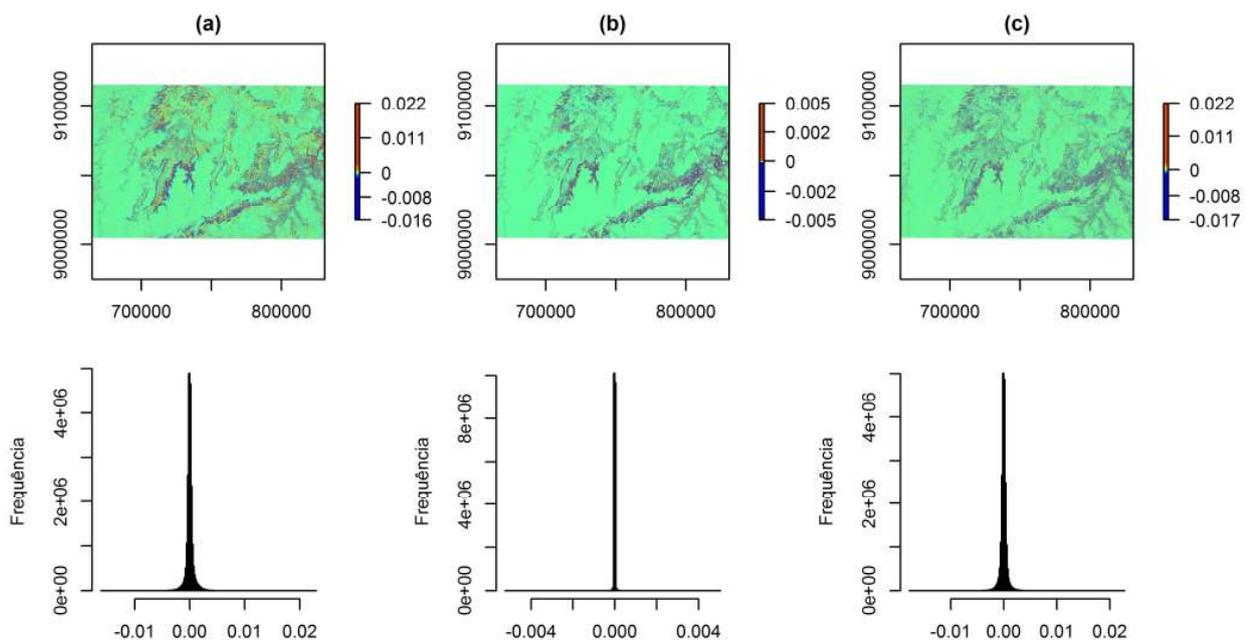


Figura 9 - Mapas de curvaturas, (a) perfil de curvatura, (b) plano de curvatura e (c) curvatura tangencial.

5.2.6 Fluxo Acumulado

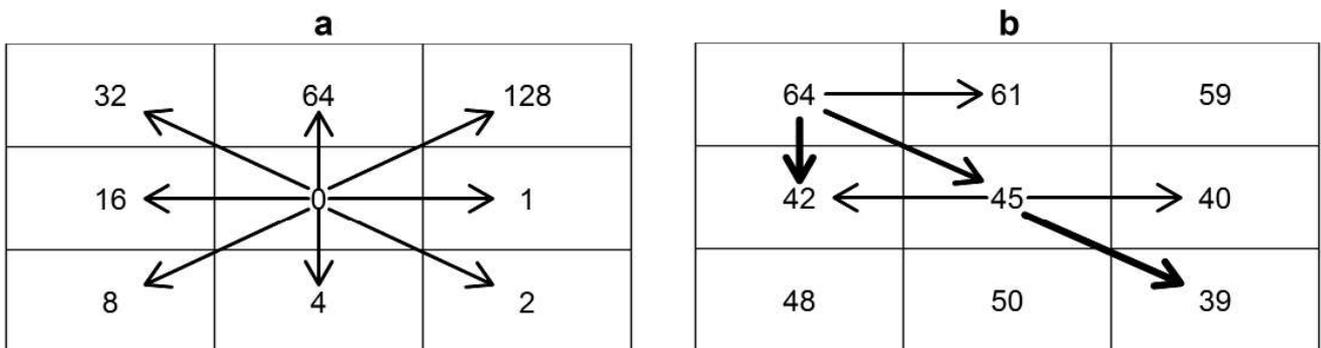
O fluxo acumulado, também conhecido como área de contribuição de fluxo, consiste numa área potencial de escoamento (*runoff*) de uma determinada região de interesse, representada por uma área plana que descreve uma extensão espacial de coleta na qual a precipitação de chuva ou neve pode ser agregada (Erskine *et al.*, 2006; Hengl & Reuter, 2009). Este parâmetro é utilizado em diversos modelos hidrológicos sendo útil na predição de áreas fontes de escoamento de água, saturação, desenvolvimento e erosão de solos e cobertura da vegetação (Moore *et al.*, 1991). Os mapas gerados apresentam rotas de escoamento de um fluido sobre a superfície do MDE. Há vários algoritmos que calculam a rota do fluxo e fundamentam-se na direção do gradiente do terreno. Neste trabalho o processamento do fluxo acumulado se deu através da biblioteca `dynatopmodel` utilizando a função `upslope.area()` (Metcalf *et al.*, 2015). A referida função baseia-se no algoritmo MFD8 (*multiple flow direction algorithm* – Quinn *et al.*, 1991) que assinala o fluxo a cada célula, em seguida verifica o sentido do escoamento para as células vizinhas. O fluxo poderá seguir em uma ou mais direções no sentido do declive (Neteler & Mitasova, 2004). Considerando uma submatriz 3x3, quando um fluido é assinalado para o pixel central, o escoamento se dará para uma das 8 células vizinhas com menor altitude (ou maior declive) baseando-se no sentido direcional do caimento, em geral, representados em dígitos binários para oito quadrantes azimutais (Figura 10a). Ao se repartir, um peso é dado para cada fluxo em função da declividade e da distância da célula vizinha (Figura 10b) (Quinn *et al.*, 1991).

Após a computação das direções de fluxo, o pixel que recebeu o fluxo de uma célula soma 1, indicando que esta recebeu uma linha de fluxo, o qual aumenta progressivamente à medida que vários fluxos incidem sobre a mesma célula (Figura 11).

```
# criar mapa de fluxo acumulado
> flowacc <- upslope.area(srtm_reprojetado, log = TRUE, atb = FALSE, deg = 0.1, fill.sinks = TRUE)
```

5.2.7 Rugosidade

O termo rugosidade apresenta diferentes abordagens na literatura sendo comumente definido como a distribuição espacial da variação da elevação (Franklin, 1987) ou o desvio padrão das elevações dentro de uma área (Jordan, 2007). De modo genérico, a rugosidade mede a complexidade de um terreno, *i.e.* seu grau de ondulação (Olaya, 2009). Assim, um terreno levemente ondulado é referido como suave, ao passo que uma superfície fortemente ondulada configura um terreno rugoso. Contudo, há uma limitação nesta última afirmação, pois, como Grohmann *et al.* (2010) observam em sua análise comparativa da rugosidade em diferentes escalas, que a variabilidade espacial das variáveis geomorfométricas indicam o quão uma área é diferente de outra dentro de uma região. Logo, os dados espaciais gerados por mapas de rugosidade carecem de uma interpretação destes valores e sua localização com as características geológicas e geomorfológicas.



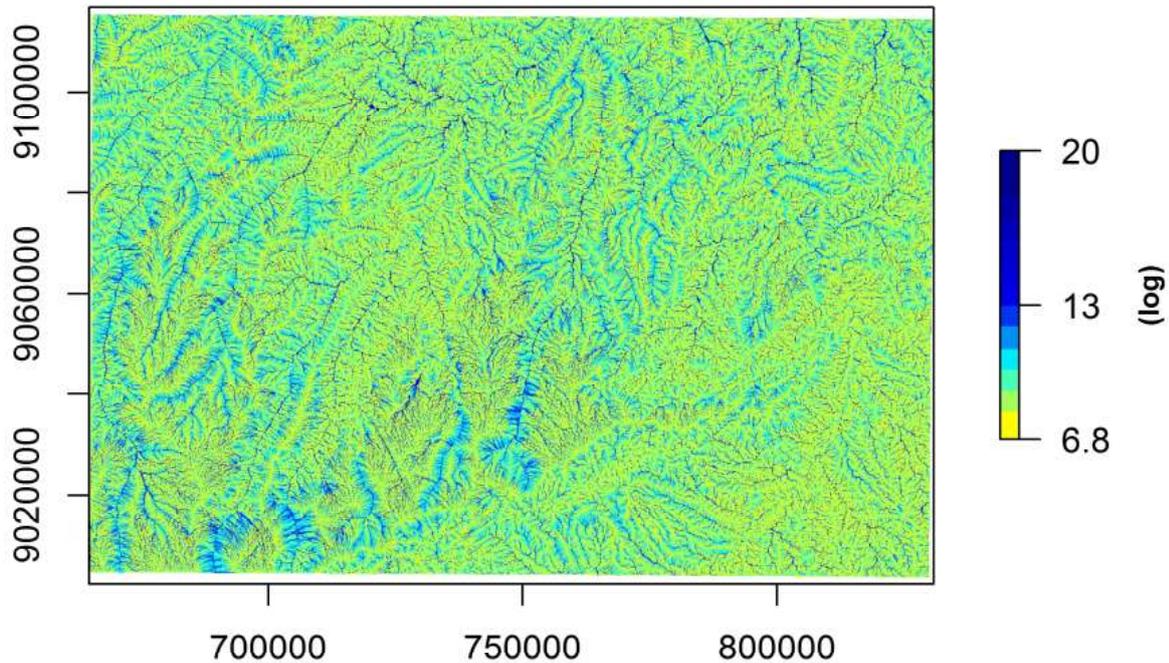


Figura 11 - Fluxo acumulado.

Aqui foram aplicadas duas abordagens para gerar mapas de rugosidade: (1) incluída no pacote `raster()` através da função `terrain()`, retorna como valor a diferença entre a altitude máxima e mínima dos 9 pixels de uma janela “andante”, aqui escolhida de 3x3 células; (2) baseia-se na razão entre a área real do pixel levando em conta a declividade do terreno e uma área plana do mesmo pixel (Equação 26), conforme Grohmann (2004).

$$\text{Rugosidade} = \frac{a \cdot c}{a^2} \quad (26)$$

onde:

$a \times c$ é a área real da superfície de um pixel e a^2 é a área plana encontrada pela resolução do pixel.

O valor da variável c é obtido a partir das relações trigonométricas entre a tangente da declividade (α) da superfície real e a resolução do pixel a (Figura 12), considerando o triângulo formado por a , b e c , em que b é o cateto oposto e c a hipotenusa do triângulo.

Assim, em termos trigonométricos, obtêm-se:

$$c = \sqrt{a^2 + b^2}$$

Ao calcular a tangente da declividade, encontra-se b :

$$\tan(\alpha) = b/a \rightarrow b = \tan(\alpha) \cdot a$$

Definidos os termos, podemos simplificar a equação da rugosidade substituindo-os:

$$\text{Rugosidade} = \frac{a \cdot \sqrt{a^2 + (\tan(\alpha) \cdot a)^2}}{a^2} = \sqrt{\frac{a^2 + (\tan(\alpha) \cdot a)^2}{a^2}}$$

$$\text{Rugosidade} = \sqrt{1 + \tan^2(\alpha)} \quad (27)$$

Para criar o mapa de rugosidade baseado na função `terrain()` do pacote `raster`, deu-se como entrada um vetor de caracteres a opção ‘`roughness`’ calculado a partir de 8 vizinhos, o qual considera a célula central:

```
> rugosidade <- terrain(srtm_reprojetado,
opt='roughness', neighbors = 8)
```

A rugosidade segundo Grohmann (2004) foi escrita em formato de equação. Antes de executar a fórmula, é necessário que o mapa raster de declividade esteja em radianos, para daí gerar o mapa final. Basta utilizar a função `terrain()` e escolher a unidade de saída para a declividade.

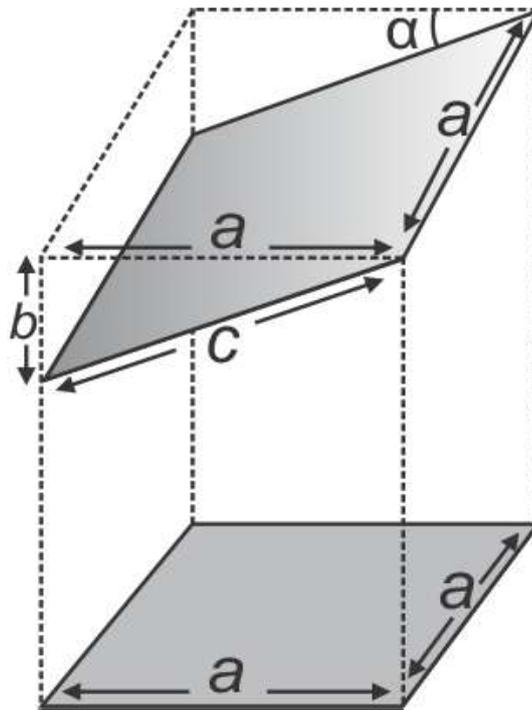


Figura 12 - Esquema de cálculo da área real de um pixel em função da declividade. Modificado de Grohmann (2004).

```
# calcular declividade com saída em radianos
> slope_rad <- terrain(srtm_reprojetado,
opt='slope', unit='radians', neighbors = 8)
# rugosidade razão area real/area plana
> rugosidade_razao <- sqrt(1+(tan(slope_rad))^2)
```

Os mapas de rugosidade para o método de maior diferença dos vizinhos e da razão área real/área plana de um pixel apresentam diferenças significativas (Figura 13). No Rstudio, ao plotar um gráfico em formato raster, sem entrada de cores, este retorna a imagem com a paleta padrão `terrain.colors`. Para melhor visualizar e comparar os mapas gerados, utilizou-se a paleta de cores em escala de cinza, cujo intervalo varia do cinza claro a branco, empregando um valor `gamma` igual a 7 a fim de melhorar o contraste para os dois mapas de rugosidade.

```
# exemplo de gráfico em escala de cinza
> plot(rugosidade_razao, col=gray.colors(100,
start=0.4, end=1, gamma = 7, alpha = NULL),
interpolate=T)
```

Os valores de saída de ambas abordagens divergem e baseiam-se na diferença dos métodos aplicados, onde a rugosidade baseada na amplitude de valores de

uma janela depende diretamente das altitudes do relevo analisado. Por outro lado, os resultados baseados na razão entre áreas do pixel real e plano possuem valores mínimo de 1, quando o pixel da área real possui gradiente nulo, e valores maiores que 1 quando a área real aumenta com o gradiente computado.

Na abordagem da amplitude de altitude observa-se que a rugosidade é mais sensível a diferenças mínimas de elevação. É possível identificar que na Figura 13(a), mesmo nos topos planos, ocorrem variações da tonalidade cinza. Na Figura 13(b), os maiores valores encontram-se associados à regiões de elevados gradientes, que neste caso, estão relacionados à escarpas erosivas dos cânions e talwegues da rede drenagem. Grohmann (2004) afirma que o método da razão entre áreas permite diferenciar morfológicamente a paisagem, pois considera a forma da superfície, assim, relevos deformados não podem ser mascarados pelas diferenças de altitudes. Assim, o modelo a ser considerado depende do objetivo a ser analisado. Em trabalhos de análise de riqueza de espécies de uma determinada área, a rugosidade pelo método da amplitude, talvez seja mais recomendado, já que populações de espécies podem sofrer maior influência das variáveis morfométricas, *i.e.* a complexidade espacial aumenta a heterogeneidade ambiental e influencia na habilidade de cada espécie dispersar e competir (Dufour *et al.*, 2006).

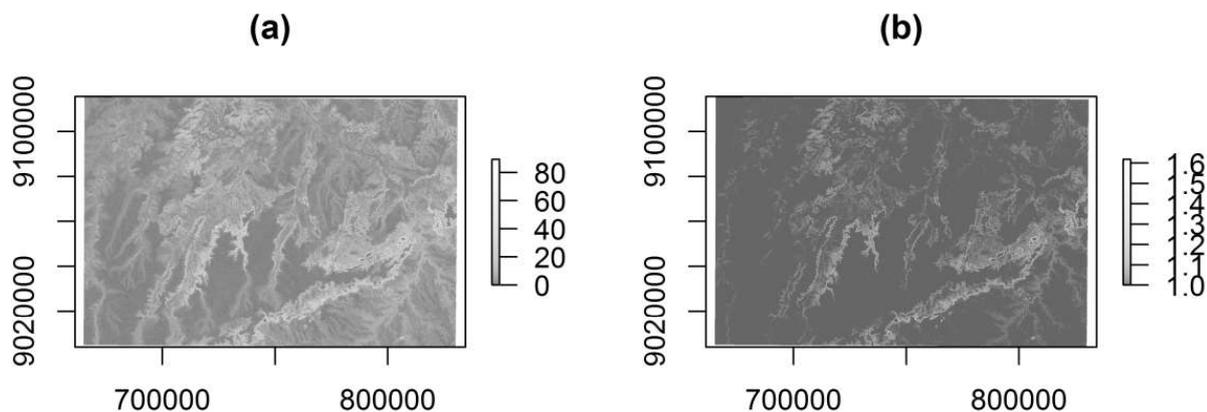


Figura 13 - Mapas de rugosidade gerado pela máxima diferença entre vizinhos (a) e pela razão de área real por uma área plana (b).

6. Conclusões

Neste trabalho foram discutidos os conceitos fundamentais de ADT, com um exemplo de aplicação com a linguagem de programação R. As variáveis geomorfológicas do relevo foram computadas utilizando uma imagem digital SRTM de 30 metros de resolução da região do Parque Nacional Serra da Capivara. Algumas funções de pacotes desenvolvidos para manipulação e extração de dados topográficos de imagens tipo raster foram empregadas. Todas as funções e scripts foram executadas e implementadas através da interface de ambiente de desenvolvimento RStudio, versão gratuita *open source*, que dispõe de ferramentas e complementos úteis para iniciantes e àqueles que ainda não estão familiarizados com a linguagem.

No ambiente de desenvolvimento, com poucas linhas de código extraiu-se os parâmetros morfométricos básicos do relevo e suas derivações. R possui uma sintaxe de fácil assimilação, por isso, permitiu uma rápida implementação dos algoritmos. Além disso, dispõe de uma grande quantidade de material educativo na internet, sendo de maioria na língua inglesa. Os passos metodológicos aplicados neste trabalho podem ser executados facilmente por qualquer usuário, mesmo iniciantes, desde que possuam os conceitos fundamentais de lógica de programação e algoritmos.

A linguagem R apresentou-se como uma poderosa ferramenta em ADT funcionando como um ambiente SIG, sendo possível implementar, desenvolver e melhorar rotinas, possibilitando a visualização e atuação do usuário em cada etapa do processo de implementação.

Referências Bibliográficas

BIVAND, R. S. Using the R statistical data analysis language on GRASS 5.0 GIS database files. *Computers & Geosciences*, v. 26, p. 1043-1052, 2000.

BÖHNER, J. & ANTONIC, A. Land-surface parameters specific to Topo-Climatology. In: HENGL, T. H. & REUTER, H. I. 2009. **Geomorphometry: Concepts, Software, Applications**. Developments in Soil Science, v. 33. Chapter 8, p. 195-226, 2009.

BROWN, R. Geomorphons: landform and property predictions in a glacial moraine in Indiana landscapes. *Catena*, v. 142, p. 66-76, 2016.

DENG, Y. New trends in digital terrain analysis: landform definition, representation, and classification. *Progress in Physical Geography*, v. 31, n. 4, p. 405-419, 2007.

DHONT, D. & CHOROWICZ, J. Review of the neotectonics of the Eastern Turkish-Armenian Plateau by geomorphic analysis of digital elevation model imagery. *International Journal of Earth Sciences (Geol Rundsch)*, v. 95, p. 34-49, 2006.

DUFOUR, A.; GADALLAH, F.; WAGNER, H. H.; GUISAN, A.; BUTTLER, A. Plant species richness and environmental heterogeneity in a mountain landscape: effects of variability and spatial configuration. *Ecography*, v. 29, 573-584, 2006.

ERSKINE, R. H.; GREEN, T. R.; RAMIREZ, J. A.; MACDONALD, L. H. Comparison of grid-based algorithms for computing upslope contributing area. *Water Resource Research*, v. 42, n. 9, W09416, 2006.

FLORINSKY, I. V. **Digital terrain analysis in soil Science and Geology**. 1st Edition. Academic Press. 432 p., 2012.

FRANKLIN, S. E. Geomorphometric processing of digital elevation models. *Computers & Geosciences*, v. 13, n. 6, p. 603-609, 1987.

GRUNSKY, E. C. R: a data analysis and statistical programming environment – an emerging tool for the geosciences. *Computers & Geosciences*, v. 28, p. 1219-1222, 2002.

IHAKA, R. & GENTLEMAN, R. R: a language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, v. 5, p. 299-314, 1996.

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (INPE).

- Disponível em: <<http://www.dsr.inpe.br/topodata/dados.php>>. Acessado em: 15/06/2016.
- GROHMANN, C. H. Morphometric analysis in geographic information systems: applications of free software GRASS and R. **Computers & Geosciences**, v. 30, n. 9-10: 1055-1067, 2004.
- GROHMANN, C. H.; SMITH, M. J.; RICCOMINI, C. Multiscale analysis of topographic surface roughness in the Midland Valley, Scotland. **Geoscience and Remote Sensing, IEEE Transactions on**, vol. PP, n. 99, p. 1-14, 2010.
- GROSSE, P.; VRIES, B. W.; EUILLADES, P. A.; KERVYN, M.; PETRINOVIC, I. A. Systematic morphometric characterization of volcanic edifices using digital elevation models. **Geomorphology**, v. 136: 114-131, 2012.
- HENGL, T. H. & REUTER, H. I. **Geomorphometry: Concepts, Software, Applications**. Developments in soil Science, v. 33. Elsevier Science, 1st edition. 796 p., 2009.
- JONES, K. H. A comparison of algorithms used to compute hill slope as a property of the DEM. **Computers & Geosciences**, v. 24, n. 4, p. 315-323, 1998.
- JORDAN, G.; MEIJNINGER, B. M. L.; van HINSBERGEN, D. J. J.; MEULENKAMP, J. E.; van DIJK, P. M. Extraction of morphotectonic features from DEMs: development and applications for study areas in Hungary and NW Greece. **International Journal of Applied Earth Observation and Geoinformation**, v. 7, p. 163-182, 2005.
- JORDAN, G. Digital terrain analysis in a GIS environment. Concepts and Development. PECKHAM, R. J. & JORDAN, G., editors, **Digital terrain modeling: development and applications in a policy support environment**, Berlin: Springer: p. 1-43, 2007.
- KOBAL, M.; CEGLAR, A.; ELER, K.; MEDVED-CVILK, B.; HONZAK, L.; SIMONCIC, P.; HLADNIK, D. On the use of R programming language in the analyses of spatial data. **Acta Silvae et Ligni**, v. 102, p. 55-62, 2013.
- METCALFE, P.; BEVEN, K.; FREER, J. Dynamic TOPMODEL: A new implementation in R and its sensitivity to time and space steps. **Environmental Modelling & Software**, v. 72, p. 155-172, 2015.
- MITASOVA, H. & HOFIERKA, J. Interpolation by regularized spline with tension: II. Application to terrain modeling and surface geometry analysis. **Mathematical Geology**, v. 25, n. 6, p. 657-669, 1993.
- NETELER, M. & MITASOVA, H. **Open Source GIS: a GRASS GIS approach**. Kluwer Academic Publishers/Springer, Boston, 2nd edition. 402 p., 2004.
- OLAYA, V. Basic Land-Surface parameters. In: HENGL, T. H. & REUTER, H. I. (eds.). **Geomorphometry: Concepts, Software, Applications. Developments in soil Science**, v. 33., p. 141-169, 2009.
- OMUTO, C. T. & VARGAS, R. R. Re-tooling of regression kriging in R for improved digital mapping of soil properties. **Geoscience Journal**, v. 19(1), p. 157-165, 2014.
- PENNOCK, D. J.; ZEBARTH, B. J.; JONG, E. Landform classification and soil distribution in hummocky terrain, Saskatchewan, Canada. **Geoderma**, v. 40, p. 297-315, 1987.
- QUINN, P.; BEVEN, P. Q.; CHEVALLIER, P.; PLANCHON, O. The prediction of hillslope flow paths for distributed hydrological modelling using digital terrain models. **Hydrological Processes**, v. 5, p. 59-79, 1991.
- R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. 2014. Disponível em: <<http://www.R-project.org/>>. Acessado em: 15/09/2016.
- RITTER, P. A vector-based slope and aspect generation algorithm. **Photogrammetric Engineering and Remote Sensing**, v. 53, n. 8, p. 1109-1111, 1987.
- RStudio Team. RStudio: Integrated development for R. RStudio, Inc., Boston, MA. 2015. Disponível em: <<http://www.rstudio.com>>. Acessado em: 15/09/2016.
- TORFS, P. & BRAUER, C. **A (very) short introduction to R**. 2014. Disponível em: <<https://cran.r-project.org/doc/contrib/Torfs+Brauer-Short-R-Intro.pdf>>. Acessado em: 15/09/2016.
- WILSON, J. P. & GALLANT, J. C. Digital terrain analysis. In Wilson, J. P. and Gallant, J. C., editors, **Terrain analysis: principles and applications**, New York: Wiley, p. 29-49, 2000.
- YOUNG, M. & EVANS, I. S. **Statistical characterization of altitude matrices by computer**. Report 5 on grant DA-ERO-591-73-G0040 (18 p.). Durham, England: Department of Geography, University of Durham. 1978.
- ZEVENBERGEN, L. W & THORNE, C. R. Quantitative analysis of land surface topography. **Earth Surface Processes and Landforms**, v. 12, p. 47-56, 1987.
- ZHOU, QIMING. Relief shading using digital terrain elevation models. **Computers & Geosciences**, v. 18, n. 8, p. 1035-1045, 1992.
- ZUUR, A. F.; IENO, E. N.; MEEESTERS, E. H. W. G. **A Beginner's Guide to R**. New York. Springer. p. 220, 2009.