

Tauana Junqueira da Cunha



tauana.cunha@usp.br • tauanajc@gmail.com

Recém graduada em Ciências Biológicas pelo IB USP. Realizei minha Iniciação Científica no Instituto Oceanográfico, sob orientação do Prof. Paulo Sumida. No Projeto “Macrofauna associada à macroalga *Dictyota* spp. (Phaeophyceae, Dictyotaceae) no Recife Sebastião Gomes e no Arquipélago dos Abrolhos, Brasil”, fiz o primeiro levantamento taxonômico da fauna fital da região de Abrolhos e estudei o padrão de distribuição temporal e espacial da macrofauna.

Atualmente estou no Laboratório de Poliquetologia do Dpto. de Zoologia (IB USP) trabalhando com a taxonomia da família Sabellidae na minha segunda IC intitulada “Sabellidae (Polychaeta) da Baía da Ilha Grande, Rio De Janeiro, Brasil” sob orientação do Prof. João Nogueira.

[Currículo Lattes](#)

Exercícios

[exec](#)

Trabalho Final

Plano A - Concluído

Criar um conjunto de funções que dê como output as coordenadas de uma lista de localidades. Em muitos trabalhos pode ser interessante apresentar a distribuição dos organismos com os quais se está trabalhando ou informar a origem do material utilizado. Entretanto, as informações disponíveis nem sempre são fáceis de serem compreendidas ou encontradas (como rótulos de organismos depositados em museus, que na maioria das vezes não são renovados e ficam com nomes desatualizados; ou artigos não tão recentes que fornecem a localização de seu material apenas com

nomes pouco conhecidos por outras pessoas).

Tendo como base uma tabela de localidades com os nomes das cidades, pretendo criar um conjunto de funções que busque esses nomes numa tabela bem completa de localidades e coordenadas, a princípio no Brasil (tabela referência de coordenadas do país baixada do GEONet Names Server, mais especificamente, desse [link](#)). Essa base inclui nomes de localidades afastadas e nomes antigos que já estão fora de uso, mas que muitas vezes são a única informação disponível. Atualmente, o usuário precisa inserir o nome da localidade, uma de cada vez no campo SEARCH>país escolhido>Name>Search database, o que retorna os registros correspondentes.

A ideia é que a função procure nessa base a partir dos dados fornecidos e, não encontrando uma correspondência perfeita, traga as melhores possibilidades para que o usuário escolha dentre elas (isso poderia ser um argumento de função com TRUE e FALSE, permitindo essa escolha ou retornando a melhor opção escolhida pela própria função). O output sairá num arquivo de texto com as coordenadas correspondentes a cada entrada. Dando certo, isso poupará um enorme esforço manual de busca.

Ainda, se for viável, pretendo utilizar esse output como base para a montagem de mapas com as distribuições das localidades. Esse [site](#) poderá ser útil.

Plano B

Automatizar uma rotina de análises multivariadas como as que utilizei na minha Iniciação Científica. A partir de dados de quantidades de espécimes e da unidade de medida do substrato, pretendo criar um conjunto de funções que generalize sua utilização para a obtenção da densidade de organismos e, a partir dessa variável, sejam rodadas várias análises multivariadas usualmente empregadas em estudos de comunidades, como nMDS e ANOSIM, passando pela criação da matriz de dissimilaridades, transformações dos dados e por testes que avaliem se os dados atendem aos pressupostos dos demais testes que serão utilizados. Os outputs serão arquivos de textos com os resultados de cada análise e gráficos bem formatados.

Comentários das propostas (Leo)

Acho a proposta A interessante. A proposta B é mais complexa e talvez de difícil execução dentro do prazo proposto. Com relação à proposta A, dois desafios seriam permitir que dentro da base de dados a função retorne caracteres aproximados (que não sejam idênticos), e um algoritmo que classifique as melhores localidades por algum critério (similaridade). Resolvendo estas duas tarefas a plotagem dos registros em um mapa é trivial e pode ser implementada facilmente.

Página de Ajuda / Help

geo.coordinates

package:nenhum

R Documentation

Coordenadas geográficas no Brasil. Mapa de distribuição.

Description:

`geo.coordinates` procura pelos nomes **de** localidades brasileiras no banco **de** dados GEONet Names Server (GNS).

É possível escolher a unidade das coordenadas geográficas que serão obtidas e, caso escolhida a decimal, a função utiliza elementos gráficos para auxiliar na desambiguação **de** localidades.

A função ainda produz um mapa final com a distribuição das localidades (para coordenadas em decimal). **Necessária** a instalação dos pacotes `sp` e `maptools`.

Usage:

```
geo.coordinates(x, output="Output", unit="dec", type=F, short=F,
generic=F, class=F, elev=F, note=F)
```

Arguments:

x caráter. **Interpretado** como nome do arquivo **de** entrada se houver apenas um elemento. **Deve** ser uma tabela em formato **de** texto com apenas uma coluna contendo cabeçalho e os nomes das localidades **de** interesse. **Ver** Examples.

Interpretado como um vetor contendo os nomes das localidades se tiver mais **de** um elemento.

output caráter. **Nome** do arquivo que será produzido. **Não** colocar a extensão. **Ver** Value.

unit caráter. **Unidade** das coordenadas geográficas. **Default:** dec (decimal). **Permite** a construção **de** mapas temporários para auxiliar na determinação das localidades e **de** um mapa final com a distribuição dos pontos **de** interesse.

Opções são dms (graus, minutos, segundos) ou mgrs (Military Grid Reference System). **Não** permitem a construção **de** mapas. **Ver** Value para interpretação dos resultados.

type lógico. **Classificação** do nome da localidade. **Se** TRUE, é inserido como coluna adicional no output. **Ver** Value para legenda dos resultados.

short lógico. **Abreviação** do nome da localidade. **Se** TRUE, é inserido como coluna adicional no output.

generic lógico. **Descrição** da localidade (como Rio, Fazenda etc). **Se** TRUE, é inserido como coluna adicional no output.

class lógico. **Classificação** da localidade. **Se** TRUE, é inserido como coluna adicional no output. **Ver** Value para legenda dos resultados.

elev lógico. **Altitude** em metros. **Se** TRUE, é inserido como coluna adicional no output.

note lógico. **Observações** do banco **de** dados sobre a localidade. **Se**

TRUE, é inserido como coluna adicional no output.

Details:

É recomendada a utilização de `unit="dec"` (unidade decimal das coordenadas) para que seja possível a construção de mapas temporários que ajudem na identificação das localidades. Para tal, os pacotes `sp` e `maptools` devem ser previamente instalados.

Por padrão, os argumentos `type`, `short`, `generic`, `class`, `elev` e `note` não são considerados na função. De acordo com o interesse nas informações, uma coluna para cada um deles pode ser adicionada inserindo TRUE nos respectivos argumentos.

`geo.coordinates` salva o banco de dados baixado do GEONet Names Server e os shapefiles do mapa do Brasil numa pasta chamada "zip" automaticamente criada dentro do diretório de trabalho. Manter a pasta "zip" torna a função mais rápida para utilização consecutiva.

Value:

`geo.coordinates` retorna um **data.frame** com os seguintes componentes nas colunas:

`original` nomes originais das localidades (fornecidos no argumento `x`)

`full` nomes como encontrados no banco de dados GNS

`lat_dec` latitude em graus decimais ($\pm$ dd.dddddd); sem sinal (+) = Norte, sinal negativo (-) = Sul.

`long_dec` longitude em graus decimais ($\pm$ dd.dddddd); sem sinal (+) = Leste, sinal negativo (-) = Oeste.

`lat_dms` latitude em graus, minutos e segundos ($\pm$ ddmmss); sem sinal (+) = Norte, sinal negativo (-) = Sul.

`long_dms` longitude em graus, minutos e segundos ($\pm$ ddmmss); sem sinal (+) = Leste, sinal negativo (-) = Oeste.

`coord_mgrs` coordenadas no sistema militar, sistema alfa-numérico de valores únicos para cada localidade do globo.

`type` Classificação do nome da localidade. **Legenda:**

C = Conventional

N = Approved

NS = Non-roman script

D = Unverified

DS = Unverified non-roman script

H = Historic

HS = Historic non-roman script

P = Provisional

PS = Provisional non-roman script
 V = Variant or alternate
 VA = Anglicized variant
 VS = Variant non-roman script

short Abreviação dos nomes das localidades.

generic Descrição da localidade (como Rio, Fazenda etc).

class Classificação da localidade. **Legenda:**
 A = Administrative region feature type
 P = Populated place feature type
 V = Vegetation feature type
 L = Locality or area feature type
 U = Undersea feature type
 R = Streets, highways, roads, or railroad

feature type

T = Hypsographic feature type
 H = Hydrographic feature type
 S = Spot feature type

elev Altitude em metros.

note Observações encontradas no banco de dados sobre a localidade.

Mais detalhes em: http://earth-info.nga.mil/gns/html/gis_countryfiles.html

O **data.frame** é apresentado na tela ou pode ser salvo em um objeto. Seu conteúdo é salvo em uma tabela no diretório de trabalho em formato de texto (.txt separado por tabulação).

Se escolhido `unit="dec"`, a função também exporta para o diretório de trabalho um mapa do Brasil (em formato pdf) com