

Mariana Santos Lobato Martins



Sou formada em Oceanologia pela Universidade Federal de Rio Grande-FURG. Atualmente sou mestranda em Oceanografia, com ênfase em Oceanografia Biológica, pelo Instituto Oceanográfico da USP.

Minha dissertação não prevê a utilização do R para a análise de dados, visto que estou trabalhando com a área social da Oceanografia e metodologias qualitativas. Mas sempre gostei de estatística e do ambiente R, além de acreditar na importância deste como ambiente de trabalho livre e colaborativo, por isso estou fazendo a disciplina.

EXERCÍCIOS

[Exercícios](#)

TRABALHO FINAL

1. Proposta principal:

No Brasil, de acordo com o Censo Demográfico de 2010, a zona costeira concentra 26,58% da população (IBGE, 2011). Considerando que muitas das grandes cidades se localizam nas zonas costeiras, é de grande importância para a gestão que se conheça o ambiente da praia e suas características. Nesse sentido, usuários, poder público, gestores, oficiais da defesa civil e do resgate necessitam de meios rápidos e eficientes para monitorar o estado do mar a fim de: minimizar possíveis efeitos das ressacas conhecendo a altura de onda incidentes; entender o estágio morfodinâmico da praia que irá influenciar na morfologia (forma da praia), por exemplo: se a praia é de tombo (alta declividade) ou suave (figura 1 a), o que altera as chances de afogamento de pessoas que não são boas nadadoras; bem como prever a possibilidade das correntes de retorno. As correntes de retorno são canais de correntes que puxam os nadadores em direção ao mar aberto, muitas vezes provocando casos de afogamento (figura 1 b). Elas se formam de várias formas, podendo ser causadas por alterações rápidas na altura de ondas, como pela chegada de um swell (grandes ondas formadas em áreas remotas com ocorrência de tempestades) ou pontos de ruptura nos bancos de areia, relacionado ao estágio morfodinâmico da praia.



Para facilitar aos leigos na área de oceanografia a obterem de forma rápida informações sobre o estado do mar e monitorar o mesmo num período de 7 dias, será construída a função `monitoraR`. Está irá utilizar dados de saída do modelo WaveWatch, baixados no R, das variáveis altura de onda em águas profundas e período da onda de forma a calcular a altura de onda na praia (equação 1), a importância da maré no sistema da praia (equação 2) e o estado morfodinâmico da praia (equação 3). Dessa forma, a função irá fornecer informações de previsão da altura da onda na praia, seu período, o estado morfodinâmico e o risco de correntes de retorno associadas para cada dia.

$H_{rsa} = (g * H_{profundo})^{0.5}$ (Equação 1)

$RTR = TR / H_{raso}$ (Equação 2)

$\Omega = \text{Hraso} / \text{Ws} \cdot T$ (Equação 3)

Onde: Hrasa é altura de onda em águas rasas, g é a aceleração da gravidade, TR é a amplitude de maré, RTR é Related Tide Range, Ω é o estado morfodinâmico da praia, Ws é a velocidade de decaimento do grão de areia, T é o período da onda.

Planejamento da função

Entrada: `monitoraR(lat, long, dia.inicio, dia.final, estado.BR, grão.carac, ripcurrent = T)`

- `lat` = latitude da praia a ser investigada (classe: numeric).
- `long` = longitude da praia a ser investigada (classe: numeric).
- `dia.inicio` = data do dia de início da previsão no formato YYYY-MM-DD (classe: data).
- `dia.final` = data do dia de final da previsão no formato YYYY-MM-DD (classe: data).
- `Estado.BR` = estado do BR a ser escolhido pelo usuário, dentre as opções estão as siglas dos estados costeiros (classe: character)
- `grão.carac` = característica do grão a ser escolhida pelo usuário, dentre as opções “AGrossa”, “AMedia”, “AFina”, “AMuitoFina”
- `ripcurrent` = se o usuário quer ou não calcular o índice de risco de corrente de retorno

Verificando os parâmetros:

- `lat` é um número entre 3 N e 33 S? Caso não seja, para-se o processamento da função: “`lat` precisa ser um número entre 3 N e 33S.”
- `long` é um número e menor que 0? Se não, escreve: “`long` precisa ser um número e < 0 .”
- Diferença entre o dia de início e de final da previsão é de 7 dias? Caso negativo, escreve: “`dia.inicio` e `dia.final` devem ter intervalo de 7 dias.”
- `estado.BR` é da classe character? Se não, escreve: “`estado.BR` deve ser character.”
- `grão.carac` é da classe character? Se não, escreve: “`estado.BR` deve ser character.”

Pseudo-código:

1. Faz um download de dataframe, salva no objeto Ondas, do site WaveWatch, entre a latitude (`lat`), longitude(`long`) e no intervalo de datas (`dia.inicio` até `dia.final`) escolhidos pelo usuário, das variáveis altura de onda (`Hprof`), período de onda (`T`);
2. Cria uma nova coluna no dataframe Ondas (`Ondas$Hrasa`) onde é convertida a altura de onda do WaveWatch para a altura de onda em águas rasas - equação 1
3. Cria objeto `dif.datas` para armazenar quantos dias de diferença entre o início e o fim do usuário escolheu
4. Cria um dataframe (`Grao`) para armazenar relação entre característica do grão e a velocidade média de decantação da classe - na primeira coluna as opções de características do grão (`Grao$nomes`) e na segunda as velocidades médias (`Grao$Ws`)
5. Cria um dataframe (`MareBR`) para armazenar a relação entre a amplitude média de maré em cada estado do Brasil - na primeira coluna a sigla dos estados costeiros brasileiros (`MareBR$nome`) e na segunda a amplitude de maré (`MareBR$TR`), conforme livro Brazilian Coastal Systems
6. Cria um dataframe `Estágio.praial` que armazena as informações que relacionam RTR, K, Hraso e o nome do estágio praial (colocar link da figura da tabela 1.1 do livro Brazilian Beach Systems)
7. Cria objeto chamado RTR para armazenar o RTR, calculado através da indexação no dataframe

MareBR que retorne o valor da segunda coluna que corresponda a sigla informada pelo usuário (armazenada na primeira coluna) e a média da coluna Ondas\$Hraso do dataframe Ondas - equação 2

8. Controle de fluxo da função: se $RTR > 10$ a função para e retorna a mensagem “Praia dominada por maré, não é possível calcular Ω ”

9. Cálculo do estágio praiar por dia -equação 3:

- Cria vetores Hraso.m.dia, T.m.dia, omega e nome.estágio de NAs de tamanho dif.datas;
- Entra em um ciclo for em que i vai de 1 até dif.datas;
- Entra em um ciclo for em que j vai de 1 até dif.datas;
- Cria vetor Hraso.m.dia que armazena a média de Ondas\$Hraso por dia (cada 24 linhas de dados no dataframe Ondas correspondem aos dados de 1 dia) indexação será feita usando esse princípio - indo da linhas $((j*24) - 23) : j*24$)
- Cria objeto T.m.dia que armazena a média de Ondas\$T por dia (cada 24 linhas de dados no dataframe Ondas correspondem aos dados de 1 dia) indexação será feita usando esse princípio, - indo da linhas $((j*24) - 23) : (j*24))$
- Calcula para cada posição do vetor omega o estágio praiar do dia:
- Guarda no vetor nome.estágio o nome do estágio praiar a cada dia a partir de indexação dos valores de RTR e Ω e Hraso que estão no dataframe Estagio.praial

1. Controle de fluxo: if ripcurrent == T

- Criar o vetor ripcurrent de NAs e tamanho dif.datas
- Calcular o índice de risco de correntes de retorno utilizando as variáveis Hraso, T e TR - baseado no índice descrito em Engle et al. (2002)

2. Plot em que no eixo Y estejam Ω e ripcurrent em função de dif.datas

Saída:

- Lista contendo: Hraso.m.dia, T.m.dia, Ω , nome.estágio, risco corrente de retorno
- Plot do valor de Ω e do risco de corrente de retorno para os dias do intervalo amostral escolhido pelo usuário.

Comentários Lucas Camacho

Oi Mariana, tudo bem?

Essa parece ser uma função que está bem estruturada pra você e aplica alguns cálculos da oceanografia (se não for da oceanografia peço desculpas pela ignorância) para devolver algumas informações para o usuário. Porém, tenho algumas perguntas que podem te fazer pensar e te ajudar no “polimento” da função

- O que acontece se eu for rodar a função e colocar, por exemplo, 3j e 28K no argumento da latitude? Retorna algum erro?
- A latitude tem um valor mínimo e máximo correto? E a longitude só precisa ser um número negativo. Poderia ser -4.2000 por exemplo?
- estado.BR só precisa ser um character. O que acontece se eu escrever estado.BR = ARIZONA? A pessoa escreveu um character e vai rolar algum erro no meio da função sendo que a pessoa usando não vai ter a mínima ideia do motivo
- Meio parecido com a pergunta anterior, e se eu colocar um character escrito grao.caract = MAISOUMENOSDURO? Exemplo idiota, mas acho que você pegou a ideia.

Por fim, achei a ideia da função legal e parece bem organizado já.

2. Proposta B

Dentro da área da oceanografia, grandes partes das pesquisas realizadas necessitam de trabalhos em campo, às vezes de vários dias para serem executados, ou próximo ou na praia e até mesmo em alto mar. Por depender de condições climáticas muito variadas e dinâmicas e, ainda por cima, questões logísticas, é de grande importância que seja realizado uma boa análise do tempo nos períodos propostos para que o campo ocorra em situação de segurança para a equipe, que seja possível realizar a coleta dos dados e de forma a maximizar a obtenção dos dados em campo utilizando recurso público para a pesquisa.

Trabalhos embarcados utilizando Navios Oceanográficos, tripulação, equipe de pesquisadores custa em média: xxxxxxxxx reais por dia, sem contar os diversos equipamentos que devem ser acoplados para que sejam coletados os dados. Além de conseguir coletar dados em segurança, qualquer pesquisador que queira trabalhar no mar precisa se preocupar com a possibilidade da perda de equipamentos de custo elevado por conta de condições de tempo, sendo mais um fator relevante para ser levado em consideração no momento do planejamento. Mesmo os trabalhos que não são embarcados, necessitam de boas condições de tempo para serem executados. No caso de trabalhos que envolvam pescadores precisamos: quando embarcados junto com eles (observadores de bordo) que o tempo seja bom para a segurança de todos, quando realizadas entrevistas em terra é preciso que o pescador não tenha saído para realizar a atividade. Logo, para quase todos os projetos de pesquisa em ciências do mar é essencial que se considere as condições de tempo, em especial altura de onda, velocidade do vento, precipitação, etc. Fatores como a lua podem vir a ser importantes para a coleta de alguns organismos marinhos que tem ciclos, em geral, reprodutivos acoplados à fase da lua, bem como é de grande importância para aqueles que trabalham com pesca. O mesmo para os defesos de captura de espécies, que pode vir a ser de interesse de pesquisadores.

Planejamento da função

Entrada: `campo.maR(previsao, dia.inicio, dia.final, mes, wlim, hlim, prob.chuva, fase.lua, recurso)`

- `previsao` = data frame contendo os dados de previsão de 7 dias das variáveis de altura (m) e período de onda (s), probabilidade de chuva (%), intensidade de vento (m/s), temperatura do ar (°C).
- `dia.inicio` e `dia.final` = números entre 1 e 31, referente aos dias dos mês (classe: numeric)
- `mes` = mês escolhido pelo usuário (classe: character)
- `wlim` = limite de intensidade do vento em m/s (classe: numeric, `wlim` > 0).
- `hlim` = limite de altura de onda em metros (classe: numeric, `hlim` > 0)
- `prob.chuva` = limite de probabilidade de chuva
- `fase.lua` = fases da lua escolhidos pelo usuário, caso não deseje levar em consideração, `fase.lua` = FALSE (classe: character)
- `recurso` = espécie que o usuário vai escolher, caso não venha trabalhar com a pesca colocar `recurso` = FALSE, dentre as opções dadas na descrição da função

Verificando os parâmetros:

- `previsao` é um dataframe contendo 5 colunas? Caso não seja, para-se o processamento da função: “`previsao` precisa ser um dataframe com 5 colunas de dados.”
- `dia.inicio` e `dia.final` são números entre 1 e 31? Se não, mensagem de erro e para-se a função “`dia.inicio` e `dia.final` devem ser números entre 1 e 31”
- Verificar se `previsao` tem o mesmo número de linhas do intervalo entre `dia.inicio` e `dia.final`
- `wlim`, `temp.lim`, `hlim` são números maiores que zero? Se não, escreve: “`wlim`, `temp.lim`, `hlim` precisam ser números maiores que 0.”
- `fase.lua` e `mes` são da classe `character`? Se não, para a função “`fase.lua` e `mes` devem ser da classe `character`”
- `recurso` é da classe `character`? Se não, escreve: “`recurso` deve ser da classe `character`.”

Pseudo-código:

-Cria um data frame `calendario` com as seguintes colunas: dias do ano (`seq(1:365)`), meses do ano (`rep` (nome de cada mes), duração do mês), fases da lua (`rep` (cada fase da lua), 7) e uma coluna para cada recurso e seu período de defeso no estado de SP (`defeso` = não poder pescar = `False`);

1. Com base no argumento `mes`, `dia.inicio` e `dia.final` - indexa o dataframe `calendário` retornando apenas as informações referentes àquele mês e no período de dias escolhidos, guardando no objeto `período.campo`
2. Cria `dados.período.campo` que é formado pela junção dos dataframes `período.campo` e `previsao`, formado por 9 colunas
3. A partir do dataframe `dados.período.campo` usando `if` vamos selecionar os dias com valores acima dos valores limites fornecidos pelo usuário para `wlim`, `temp.lim`, `hlim` e `prob.chuva`, criando outra coluna em `dados.período.campo` chamada `Tempo` e atribuindo valor 1 para esses dias
4. Fazer o mesmo para os valores menores e iguais aos limites fornecidos pelo usuário para `wlim`, `temp.lim`, `hlim` e `prob.chuva` - atribuindo valor 0
5. If `fase.lua` = T selecionar os dias dentro do período escolhido (indo de `dia.inicio` ao `dia.final`) e retornar as fases da lua referente a estes dias (coluna de dados no objeto `calendário`) - criar nova coluna em `dados.período.campo` chamada `lua`
6. If `recurso` = T selecionar os dias dentro do período escolhido (indo de `dia.inicio` ao `dia.final`, de acordo com o recurso escolhido pelo usuário e retornar `FALSE` para período de defeso e `TRUE` para pesca aberta (coluna de dados no objeto `calendario`) - criar nova coluna em `dados.período.campo` chamada `defeso`
7. Calcula a media de `wlim`, `temp.lim`, `hlim` e `prob.chuva` para tempo igual a 1 e igual a 0

Saída:

- Lista contendo: as colunas `dias$dados.período.campo`, `lua$dados.período.campo`, `tempo$dados.período.campo`, `defeso$dados.período.campo` e as médias de cada variável de acordo com os valores de tempo (0 ou 1)

Comentários Lucas Camacho

Já essa função eu achei que a proposta tem fundamento porém acho que ela está fazendo muito pouco para o usuário da função. Pelo que eu entendi você cria um data.frame e vai selecionando

os dados baseado nos outros argumentos da função e retorna valores médios para o usuário.
Sugiro que você reveja toda sua ideia e o que você quer retornar para o usuário final.

Além disso, queria te atentar para a Verificação de parâmetros dessa proposta semelhante a feita na proposta anterior

- Nos argumentos que precisam ser character, o que acontece se eu colocar um character meio bizarro e inesperado? Parece bobo mas é sempre bom lembrar que uma pessoa que pegar sua função não vai saber direito como funciona e como ela pode ou não usá-la.

Finalmente, se você manter as ideias dessa forma eu aconselho que você trabalhe com a proposta A e se quiser trabalhar com a B, pense no que você quer fazer, no que quer entregar para o usuário e o mais importante, o quão complexo ou quão desafiador construir essa função pode ser pra você.

Bom, qualquer dúvida ou desespero (rs) pode mandar e-mail pra mim que eu respondo assim que possível (lucas.camacho@usp.br)

Grandes abraços

Links para o trabalho final

Acabei decidindo fazer o trabalho final desenvolvendo a Proposta Principal. Os argumentos da função e alguns passos foram modificados da proposta, para facilitar o entendimento do usuário, reduzir a chance de erros e otimizar o processamento. A função `monitoraR` baixa arquivos do modelo WaveWatch III e calcula diversos parâmetros importantes para entender a dinâmica da praia, suas características morfológicas e o risco de correntes de retorno para as datas desejadas pelo usuário, da data presente até 6 dias no futuro.

- Link para código da função: [monitoraR](#)
- Arquivo da funcao: [funcao_monitorar_finalizada.r](#)
- Link para a ajuda da função : [Help](#)

From:
<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2019:alunos:trabalho_final:mariana.santos.martins:start

Last update: 2020/08/12 06:04