

1 PROPOSTA A

1.1 Motivação

O conhecimento do regime hidrometeorológico sobre uma bacia hidrográfica é fator determinante em estudos que visam o entendimento da dinâmica hidrológica nessas áreas, principalmente em áreas de cabeceiras para o melhor gerenciamento dos recursos hídricos da região. Porém, uma das grandes dificuldades em se realizar estudos concisos está associada a qualidade dos dados pluviométricos, devido a falhas na série histórica, assim, o objetivo desta proposta é criar uma função para o preenchimento das falhas presentes nas séries históricas e a verificação da consistência dos dados, garantindo assim uma melhor qualidade para esses dados pluviométricos.

1.2 Descrição da função

A função filtrará os ruídos do sensor e detectará os erros, como dados discrepantes e travamentos (function filter), para depois realizar o preenchimentos dos dados (function fill).

1.3 Function filter()

1.3.1. Dados de entrada

A função exigirá a entrada de arquivos .txt de cada estação do conjunto. (usar apply para aplicar para todos os tempos t, assim a função calculará o mae (erro) de cada estação)

1.3.2 Filtrar o ruído do sensor

A precisão dos sensores de precipitação utilizados é de 0.1 mm. if dado <0.2 dado = NA

1.3.3 Detectar erros do sensor

Detectará erros do sensor, como dados discrepantes e travamentos através da formula do erro médio absoluto, que é a média de todos os erros absolutos.



Onde:

n = número de estações meteorológicas;

Σ = símbolo de somatória;

$|x_i - x|$ = erros absolutos;

Seguindo as etapas: primeiramente encontrará todos os erros absolutos, $x_i - x$; somará todos eles; e dividirá pelo número de erros (p.e. se tiver 12 medidas, dividirá por 12).

Calcular a “P_mean” média espacial dos dados de chuva em um tempo t da estação i (p.e. $0+0+3+0+0/5 = P_mean$ próxima de 0);

Calcular o “Epi” erro da precipitação para estação i (calcula o erro da precipitação para cada estação, no exemplo anterior será muito pequeno em relação a média);

Calcular o “MAE” espacial (erro médio absoluto), que é erro médio de todas as estações para o tempo t 9;

If Epi (valor a ser preenchido) > 20% (valor limiar a decidir

em relação a distribuição dos erros)

$E_{pi} = NA$

I.e. Se o erro em relação a média é superior a 20% do erro médio absoluto de todas as estações para o tempo t , então é uma falha = NA.

1.4 Function fill ()

A função fill se justifica, pelo fato de que, uma vez que as estações estão com os dados filtrados, os dados presentes são dados consisos, logo, preencher as mais próximas com os dados filtrados, é coerente.

1.4.1 Dados de entrada

A função exigirá a entrada dos dados de saída da função filtrar com a latitude e longitude de cada estação.

1.4.2 Organizar estações

Ranquear as estação por distância em relação a estação a preencher.

1.4.3 Preenchimento

Preencher com os dados da estação mais próxima, se a mais próxima não tiver dados disponíveis, utiliza da segunda mais próxima e assim por diante.

Em suma, a função calculará o Erro médio das estações, ou seja, explicará a variação de todas as estações ao redor da média, se a estação i tem um erro muito diferente do erro das outras estações, a estação i tem um problema, logo, se esse erro for maior que o limiar determinado, ela será NA. Após esse processo, será realizado o preenchimento das falhas com dados das estações mais próximas.

2 PROPOSTA B

2.1 Motivação

A crescente demanda por informações hidrológicas das bacias hidrográficas, tais como a quantidade de água disponível nessas áreas e sua variação temporal, tem sido também cada vez mais utilizadas para estudos de planejamento e gestão dos recursos hídricos, assim, os dados de vazão dos rios devem ser monitorados e fornecidos da melhor forma e qualidade possíveis. Os principais bancos de dados brasileiros que fornecem dados de quantidade de água nas bacias hidrográficas, os fornecem principalmente na unidade cota (nível de água). Porém, não é apenas o nível da água que influencia a vazão, mas também a declividade do rio e a forma da sua seção. Assim, o objetivo dessa função é a através de uma equação de curva-chave, que é uma relação nível-vazão em uma determinada seção de um rio, automatizar a conversão dos valores de cota em vazão para um determinado rio.

2.2 Descrição da função

Criação de uma rotina, primeiramente verificando a consistência dos dados de cota, ou seja, a presença de valores “zeros” e/ou “NA” e substituição pela média dos 6 valores anteriores e dos 6 posteriores, para depois aplicação da equação de curva-chave, onde o formato de saída dos dados será na unidade de em m³/s.

2.3 Function curvachaveQ2

2.3.1 Dados de entrada

A função exigirá a entrada de arquivo .csv do nível de água do local.

2.3.2 Verificação da consistência dos dados Verificar a presença de “zeros” (visto que o nível da água de um rio não pode ser “zero”, exceto em casos x) e NA.

2.3.3 Aplicar a equação da curva-chave construída para determinado rio

$$Q = 3692,5x^3 + 561,63x^2 - 4,1881x + 0,0029$$

Onde:

Q = dado de saída em m³/s;

X = dado de entrada;

Oi Mariane,

gostei da sua proposta A e acho que você pode prosseguir com ela, só tenho algumas sugestões:

1) deixar a função mais geral pra aceitar dados de qualquer tipo, toda a sua lógica ainda faria sentido para dados não meteorológicos. 2) é melhor a função aceitar um objeto como um data.frame ou uma matriz do que um arquivo de dados. 3) juntar a filter e a fill em uma função só, mas com um argumento para o usuário especificar se ele quer que o fill seja feito. Se ele não quiser, a função troca todos os dados incertos por NA. 4) ao invés de dados de coordenadas para a parte do fill, não seria mais prático o usuário já fornecer pra função uma matriz de distâncias? Assim o usuário escolhe o critério de distância que ele quiser (espacial, temporal ou outra coisa) e a função só segue o que ele pediu.

Mas, apesar das sugestões de mudança, achei uma ideia legal e factível. Pode prosseguir tranquila.

—[Danilo G Muniz](#)

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

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2017:alunos:trabalho_final:mchitolina:propostas 

Last update: **2020/08/12 06:04**