Propostas de trabalho final

Proposta A

Criar uma função que, na entrada de um data.frame pelo usuário, especificando uma das variáveis como resposta, calcula e retorna o melhor modelo linear ajustado baseado no critério de informação de Akaike, partindo do modelo completo com todas as variáveis e suas interações.

Justificativa

Na análise de modelos lineares, o número de termos (variáveis e interações entre elas) aumenta exponencialmente conforme o número de variáveis observadas. Essa relação pode dificultar o trabalho do analista, pois além do grande numero de termos, a retirada de um deles pode mudar a significância dos outros termos, assim, faz sentido automatizar o processo de comparação, baseando-se no valor do critério de informação de Akaike de segunda ordem (AICc) (Burnham, 1998):

onde n é o número de observações, sigma-chapéu^2 é a soma dos quadrados dos resíduos dividida por n, e K representa o número de parâmetros do modelo +1.

Pseudo-código

Entrada = `lmanal(dataf,var.res=colnames(dataf)[1])`

- `dataf` = Data.frame de entrada pelo usuário
- `varres` = Variável resposta, como padrão a primeira variável do `dataf`

Premissas e manipulações

- `dataf` é data frame
- `na.omit(dataf)`
- Caso `lmanal(dataf,var.res!=colnames(dataf)[1]`
  - Reorganizar o data frame para que a variável resposta esteja na primeira coluna

Montando o modelo completo

- Extrair os nomes das colunas (variáveis) com o `paste0()` em um string
- Montar a formula do modelo completo em string: `var1~var2*var3*…*varn` com o `paste0()`
- Calcular o modelo completo com `lm()`

Comparando modelos

- O modelo completo é agora o modelo atual (que servirá de comparação)
- Retirar os termos do modelo atual com a função `attr(terms(),“terms.labels”)`
- Utilizar a função `update()` para retirar o termo de ordem mais alta, criando um novo modelo a

ser testado
- Comparar os valores de AICc do modelo atual e do modelo sendo testado
- Caso o valor de AICc do modelo sendo testado seja menor que o AICc do modelo atual, o modelo atual é agora o modelo sendo testado
- Continuar o processo de avaliação com o modelo sem a variável de maior ordem, até que se tenha avaliado todos os termos

Saída

- Retornar lista com:
  1. Modelo com o menor valor de critério de Akaike
  2. Termos significativos, seus respectivos coeficientes e p-valores
  3. Valor de R^2 do modelo atual

Referências

- Burnham KP, Anderson DR. Practical use of the information-theoretic approach. InModel Selection and Inference 1998 (pp. 75-117). Springer, New York, NY
- Burnham KP, Anderson DR, Huyvaert KP. AIC model selection and multimodel inference in behavioral ecology: some background, observations, and comparisons. Behavioral ecology and sociobiology. 2011 Jan 1;65(1):23-35.
- Wagenmakers EJ, Farrell S. AIC model selection using Akaike weights. Psychonomic bulletin & review. 2004 Feb 1;11(1):192-6.

Comentários Lucas Freitas

Boa proposta. Você está automatizando uma tarefa facilmente executável e esse é o proposto pela disciplina. No entanto seguem ainda algumas considerações:

Acredito que deva ter ocorrido um erro na sua proposição de entrada de dados. Entrada = lmanal(dataf,var.res=colnames(dataf)[1] = dataf[1])

A expressão atribuída a var.res possui dois sinais de igual. Acredito que, como o descrito abaixo da expressão, você apenas quis dizer que é um argumento, que se não imputado, deveria ser igual ao nome da primeira categoria no seu dataframe. Deste modo a expressão deveria ser:

Entrada = lmanal(dataf,var.res=colnames(dataf)[1])

Na parte do “Comparando os modelos” Muito cuidado. Pelo que eu entendi vc gostaria de automatizar o processo de uma forma que o índice de Akaike seja calculado para combinação de variáveis possível. Eu sugiro que vc:

i) crie um vetor que contenha em suas posições todas as combinações possíveis em strings,

ii) passe por esse vetor com a ajuda de um loop calculando o índice de akaike para cada uma das combinações,

iii)guardar o valor do índice em um vetor de mesmo tamanho na posição respectiva.

Com esse vetor calculado vai ser mais fácil evitar erros relacionados ao tamanho do loop necessário. Além de facilitar para saber qual índice é referente a quais parâmetros Deste modo você pode facilmente achar qual dos elementos no vetor calculado possui o índice de interesse

(Esse método é apenas uma sugestão dentro dos casos possíveis, sinta se livre para realizar o código da forma como achar melhor)

Não se esqueça de nomear adequadamente os elementos da lista e nomear o que está sendo apresentado dentro de cada elemento

Proposta B

Criar uma função que faça diversos gráficos de pizza (pie charts) em posições ordenadas em um plano cartesiano. O ângulo interno das fatias representa o tamanho de cada população em relação ao todo, e o raio da pizza é em função da soma das populações de uma amostra em relação as outras.

Justificativa

Em oceanografia é comum a exibição de dados em planos cartesianos espaciais, com o eixo das ordenadas representando a profundidade (água ou sedimento) e o das abscissas representando alguma distância horizontal. Para a exibição de dados de granulometria ou populações biológicas, em que cada parcela representa uma proporção do todo, faz sentido o uso de gráficos de pizza, nos quais o raio do gráfico representa a soma das populações ou qualquer variável contínua relacionada a distribuição das populações.

Pseudo-código

Entrada = `piexy(x,y,pop,rad=sum(pop(x,y)))`

- `x`= posição horizontal da amostra
- `y`= posição vertical da amostra
- `pop`= populações amostradas
- `rad`= raio das pizzas, como default a soma das populações em cada amostra

Premissas

- `length(x) == length(y)`
- `pop` é um data.frame com colunas `x,y,pop1,pop2,…,popn`

Cálculos

- Normalizar o atributo `rad` na variável `rnorm`, para tamanhos adequados a visualização de múltiplos gráficos no mesmo plano cartesiano
- Normalizar `x` e `y` em função dos valores de margem (`mar=c()`), nas variáveis `xnorm` e `ynorm`, respectivamente

Gráfico

- Para cada valor de `xnorm`, plotar as pizzas na posição `ynorm` com atributo `radius=rnorm`

Comentários Lucas Freitas

Essa função infelizmente não se enquadra nos pré requisitos por ser muito simples(e.g não possuir algum tipo de controle de fluxo). Caso deseje seguir com essa proposta eu sugiro que vc de mais liberdade ao usuario. Talvez colocando opcoes esteticas para os graficos.

Peço desculpas se entendi algo errado . Qualquer coisa estou a disposição.

Email: rodriguesdefreitaslucas@gmail.com Boa sorte.