IGCE/UNESP
Pós-Graduação em geografia
Análise Quantitativa de Dados Ambientais

Professor: Thiago S. F. Silva
E-mail: tsfsilva@rc.unesp/br
2º Período de 2015

Exercício 2 – Regressão Simples: ajuste, diagnóstico, remediação e validação

Neste exercício, praticaremos a análise de regressão simples, cobrindo o conteúdo mostrado nas aulas 4 e 5.

Os exercícios deverão ser necessariamente realizados utilizando a linguagem R. Para cada exercício, serão dadas algumas dicas, mas "quebrar a cabeça" é parte do processo de aprendizado de qualquer linguagem/software. Algumas informações úteis:

- O texto em `cinza` indica o código em R. `#` identifica comentários (não são executados).
- O texto de ajuda sobre qualquer função do R pode ser obtido digitando-se "?" seguido do nome da função (ex: `?mean`).
- Este site possui vários exemplos de gráficos feitos em R, com o código equivalente: http://gallery.r-enthusiasts.com/
- Google is your friend.

A) Importando e inspecionando os dados no R:

Os dados usados neste exercício compreendem os dados simulados da relação entre Índice de Área Foliar (IAF, ou LAI em inglês) e o Índice de Vegetação da Diferença Normalizada (NDVI). Os dados se encontram no formato CSV. Nosso objetivo é ajustar e avaliar um modelo que seja capaz de prever o LAI em função do NDVI. Com um modelo como esse, seria possível estimar o LAI de maneira espacializada, através de uma imagem de satélite.

```
# Carregando e inspecionando os dados

setwd("~/Dropbox/ESTECO/Exercicios/Ex3_Reg_Anova")
# Não esqueçam de mudar o caminho para o seu diretório de trabalho

ndvi.lai <- read.csv('ndvi_lai.csv')

str(ndvi.lai)
```

B) Análise Exploratória

PERGUNTA 1: O que é LAI? O que é NDVI? Use o seu Google-fu, e proponha um modelo conceitual da relação entre as duas variáveis (não precisa escrever uma tese, 2-3 parágrafos já bastam)

PERGUNTA 2: Prepare um gráfico de dispersão dos dados, nomeando seus eixos adequadamente. O que você pode dizer sobre a relação entre LAI e NDVI? Ela está de acordo com o seu modelo conceitual?

PERGUNTA 3: Plote os histogramas de NDVI e LAI. Não esqueça de ajustar o valor dos seus `breaks` adequadamente, e nomear os eixos. Os valores de NDVI parecem ser normalmente distribuídos? E os de LAI? É importante que o NDVI e/ou o LAI tenham distribuição normal? Por que?

PERGUNTA 4: Ajuste um modelo linear simples, plote os resíduos vs. valores preditos, e faça o Q-Q plot dos resíduos. Com base nesses gráficos e nos gráficos anteriores, quais pressuposições do modelo de regressão linear simples serão provavelmente violadas se ajustarmos um modelo de LAI em função do NDVI, com os dados originais?

Dicas de comandos:

```
residuals(modelo) # extrai resíduos
predict(modelos) # extrai Y_hat
abline(h= 0,col='red’) # plota uma linha horizontal vermelha no zero
qqnorm(residuals(modelo)) # qqplot
qqline(residuals(modelo)) # adiciona a linha de referencia ao qqplot
```

C) Ajustando o Modelo

PERGUNTA 5: O que podemos fazer para linearizar a relação entre X e Y? Sugira um método, descreva-o, e aplique aos dados. Avalie o resultado através de análise gráfica. As violações foram resolvidas/minimizadas?

Dicas de comandos:

```
log10(), srqt(), ^ # potência
boxcox() # pacote MASS
which()  # se for usada a função boxcox
```

PERGUNTA 6: Ajuste o modelo linear entre NDVI e LAI após a aplicação do método de linearização escolhido na pergunta 4, e explique os resultados mostrados por `summary(m)` e `anova(m)`, onde `m` é o modelo ajustado.

Dicas de comandos: `lm(), anova()`

D) Diagnóstico do Modelo

PERGUNTA 7: Faça a análise diagnóstica do seu modelo, analisando o histograma ou boxplot dos resíduos, o gráfico dos resíduos versus $\hat{Y}$, e um gráfico QQ plot dos resíduos versus uma distribuição normal. Discuta sua análise em relação às pressuposições do modelo de regressão.

Dicas: `hist()` e `boxplot()`
`predict()` e `residuals()`
`qqnorm()` e `qqline()`

PERGUNTA 8: Faça a avaliação diagnóstica da influência de cada observação sobre o modelo, usando as medidas DFFITS, Distância de Cook e DFBETAS. Há algum ponto que seja notavelmente influente? Se sim, que ponto é esse?

Dicas: `dffits()` , `cooks.distance()` , `dfbetas()`, `which(x == max(x))`
O resultado de `dfbetas()` possui uma coluna para cada coeficiente

E) Avaliação e Validação do Modelo

PERGUNTA 9: Calcule os intervalos de confiança, com nível de significância de 5%, para os coeficientes β_0 e β_1, e para E(LAI) e LAI_{novo} quando NDVI = 0.6

Dicas: `summary(lm())`

```
confint()
newdata <- data.frame(nomecoluna = valor)
predict(modelo,newdata,interval='confidence')
predict(modelo,newdata,interval='predict')
# newdata deve ser uma data.frame com uma coluna que tenha o mesmo
# nome que a coluna de X usada originalmente no ajuste do modelo
# como queremos só um valor, essa df terá 1 coluna x 1 linha
# para predições de multiplos novos valores, é só fornecer multiplos
# X novos
```

Pra quem quiser calcular manualmente, o valor t (1-alpha, n-2) pode ser calculado usando:

```
# qt(1-alpha/2,df)
# exemplo
qt(0.975,30) # t para alpha = 0.05 (5%, 1-alpha), com 30 graus de liberdade
```

PERGUNTA 10: Ajuste um novo modelo, excluindo a observação mais influente identificada pelas medidas de influência na pergunta 8. Calcule os novos intervalos de confiança para β_0 e β_1. A diferença é **cientificamente** significativa? O que **você** acha?

Dicas: `lm(y ~ x, data  = df[-1,])`

PERGUNTA 11: Agora que o seu modelo está pronto, só falta estimar a sua precisão na estimativa do LAI a partir do NDVI derivado de imagens Landsat. Mostre a distribuição dos erros e estime o erro médio quadrático (RMSE), e das suas estimativas utilizando o método *jacknife.* Utilize o código abaixo:

```
# df é o nome da data.frame com os seus dados. Substitua de acordo
# Y é o nome da variável resposta. Substitua de acordo
# X é o nome da variável explicativa. Substitua de acordo

n = dim(df) # número de repetições do jackknife/loocv = número de obs.

dif <- vector(n, mode='numeric') # cria um vetor vazio para guardar os
resultados

for (i in c(1:n)){
            m.menos.i <- lm(Y ~ X, data = df[-i,])
            Yi_pred <- predict(m.menos.i,df[i,])
            dif[i] <- Yi_pred-df$Y[i]
            }

#Função para calcular o RMSE:
rmse <- function(x) sqrt(mean(x^2))
rmse(dif)
```