Teste 07

Grupo 03

Exercício 4.26

Este conjunto de exercícios é referente ao banco de dados "CDI" utilizado em exercícios anteriores. Aqui, voltamos nossa atenção à regressão do número de médicos ativos utilizando a variável "total da população" como preditora.

Item a

Devemos construir um intervalo conjunto de Bonferroni para ambos β_0 e β_1. Utilizando a resolução do exercício 1.43, teste 05, temos

```
betas.ex143
```

```
##             [,1]        [,2]
## [1,] -110.63478 0.002795425
## [2,]  -95.93218 0.743116444
## [3,]  -48.39485 0.131701189
```

Lembre-se que o intervalo -conservativo- conjunto dado pela desigualdade de Bonferroni é

$$\hat{\beta}_i \pm t_{n-2;alpha/4}\sqrt{\widehat{V(\widehat{\beta}_i)}}.$$

Logo, o código a seguir nos permite obter os intervalos para os intercepto e coeficiente angular, respectivamente.

```
m = length(betas.ex143[1,])
alfa = 0.05

IC.bonf = matrix(nrow = m, ncol = 2)
colnames(IC.bonf) = c("Lower Bound", "Upper Bound")
rownames(IC.bonf) = c("Intercept", "Slope")

for (i in 1:m)
  IC.bonf[i,] = betas.ex143[1,i] + c(-1,1)*summary(modelo.ex143[[1]])$coef[i,2]*qt(1-alfa/(2*m), df = su
```

Que são

```
##           Lower Bound Upper Bound
## Intercept -188.783269  -32.486285
## Slope        0.002687    0.002904
```

Item b

Neste item, é sugerido que $\beta_0 = -100$ e $\beta_1 = 0.0028$. Ao observar os limites construidos em a), podemos concluir que o intervalo suporta sim esta proposta, pois -100 está entre $[-188.783269, -32.4862855]$ e 0.0028 está entre $[0.0026866, 0.0029042]$.

itens c-d

Gostariamos de estimar o número esperado de médicos para condados de tamanho de tamanhos 500, 1000 e 5000, ou seja, é um problema de predição. Para uma boa estimação, criamos intervalos de confiança para os 3 estimadores pontuais das médias das distribuições. Entretanto, para construir estes intervalos, temos dois possíveis procedimentos, sendo eles Bonferroni e Working-Hotelling. Assim, vamos primeiro avaliar cada método, escolher o mais eficaz e só então construir o intervalo para o resposta média para cada condado.

Primeiramente, note que os métodos são bem parecidos

1. Bonferroni

   $$\hat{Y}_h \pm B\sqrt{\widehat{V(\widehat{Y_h})}}$$

   onde

   $$B = t(n-2; alpha/(2g))$$

2. Working-Hotelling

   $$\hat{Y}_h \pm W\sqrt{\widehat{V(\widehat{Y_h})}}$$

   onde

   $$W^2 = gF(1-alpha; g, n-2)$$

   g é o número de estimadores pontuais para os quais queremos criar o intervalo conjunto e a variância do estimador de predição de média, $\hat{Y}_h$ é

   $$V(\hat{Y}_h) = \sqrt{MSE}\sqrt{\frac{1}{n} + \frac{(X_h - \hat{X})^2}{S_{xx}}}$$

   .

Desta forma, para obtermos o melhor intervalo basta comparar os quantís W e B e ver qual é o menor, pois a única diferença entre as duas técnicas é o quantil. Segue o algorítmo que obtém estes valores.

```
alfa=.1
g=3
(B = qt(1-alfa/(2*g) ,df=summary(modelo.ex143[[1]])$df[2]))
```

```
## [1] 2.134781
```

```
(W = sqrt(g*qf(1-alfa ,g ,summary(modelo.ex143[[1]])$df[2])))
```

```
## [1] 2.507788
```

Como 2.1347811 é menor 2.5077875, concluimos que o intervalo de Bonferroni é melhor. Assim, vamos construir um intervalo para cada nível de condado onde a população total satisfaz a condição pedida.

Antes de calcular o diretamente intervalo vamos calcular, por partes, as predições e as variâncias dos estimadores pontuais das médias. Primeiramente as predições, como segue abaixo

```
X_h = c(500,1000,5000)
beta.ex426 = lm(num_Phys ~ Total_Pop)
y.hat = rep(0,length(X_h))
for (i in 1:length(X_h)){
  y.hat[i] = beta.ex426$coefficients[[1]] + beta.ex426$coefficients[[2]]*X_h[i]
}
y.hat
```

```
## [1] -109.23706 -107.83935  -96.65765
```

Agora as variâncias

```
n=nrow(CDI)
v_y.hat = rep(0,length=length(X_h))
MSE2 = summary(beta.ex426)$sigma
Sxx  = sum((Total_Pop - mean(Total_Pop))^2)
for (i in 1:length(X_h)) {
  v_y.hat[i] = MSE2*sqrt((1/n + (X_h[i]-mean(Total_Pop))^2/Sxx))
}
v_y.hat
```

```
## [1] 34.73280 34.71958 34.61430
```

E por fim os intervalos

```
IC.ex426c = matrix(nrow = length(X_h),ncol = 2)
rownames(IC.ex426c) = c("X_1","X_2","X_3")
colnames(IC.ex426c) = c("lower","upper")
for (i in 1:length(X_h)) {
  IC.ex426c[i,] = y.hat[i] + c(-1,1)*B*v_y.hat[i]
}
IC.ex426c
```

```
##         lower     upper
## X_1 -183.3840 -35.09015
## X_2 -181.9581 -33.72064
## X_3 -170.5516 -22.76370
```

Exercício 6.28

Item a

Ao fazer o gráfico de ramo e folha para cada variável preditora (utilizando escala 20, para termos maior precisão), podemos perceber que a população está fragmentada em cidades de areás menores, enquanto que nas maiores áreas não há tanta concentração, ou seja, este é um pais não muito povoado. Os fatos que suportam esta ideia são as grandes concentrações de massa nos "topos"/inicios de todos gráficos, exceto para a variável "área do território", que parece estar relativamente despersa ao longo dos níveis.

Item b

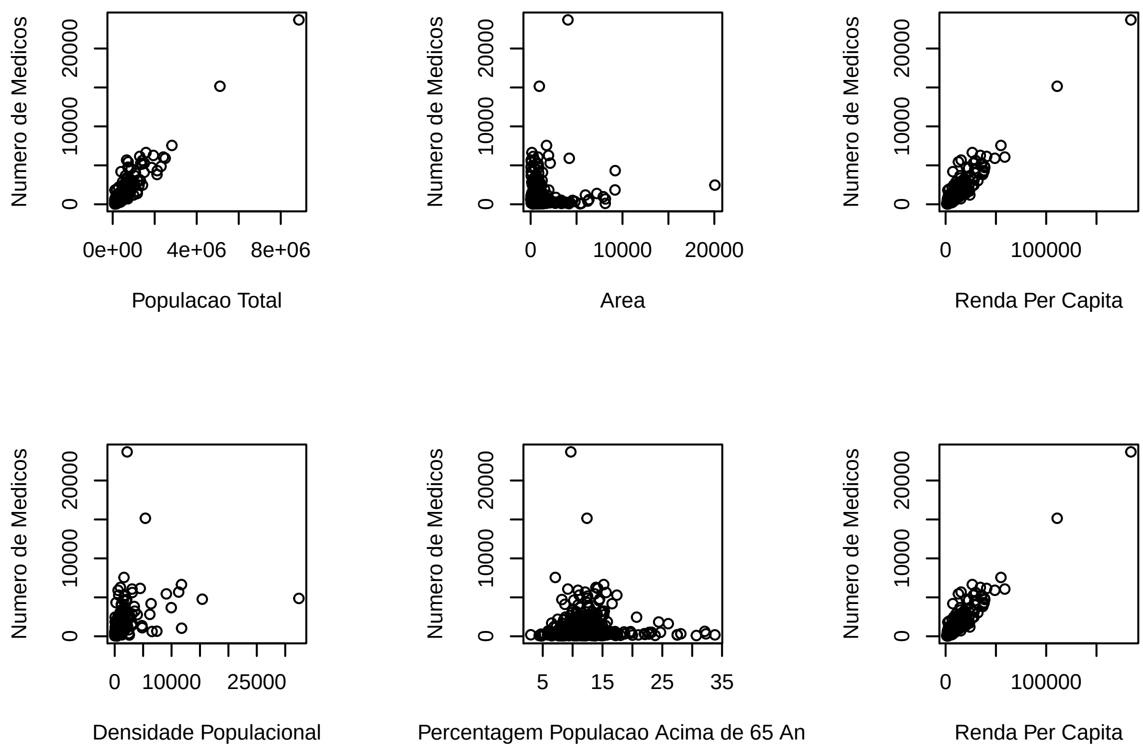
Para construirmos a matriz de dispersão para as variáveis desejadas foi utilizado o pacote "GGally" e em seguida o comando ggpairs, como segue abaixo.

```
modelo.ex628 = vector("list", length = 2)
dens_Pop = Total_Pop/Land_Area

modelo.ex628[[1]] = lm(num_Phys ~ Total_Pop + Land_Area + Total_Person_Inc)
```

```r
modelo.ex628[[2]] = lm(num_Phys ~ dens_Pop + Pct_65Plus + Total_Person_Inc)

par(pty="s")
par(mfrow = c(2,3))
plot(x = Total_Pop, y = num_Phys, xlab = "Populacao Total" , ylab = "Numero de Medicos")
plot(Land_Area,num_Phys, xlab = "Area", ylab = "Numero de Medicos")
plot(Total_Person_Inc,num_Phys, xlab = "Renda Per Capita", ylab = "Numero de Medicos")
plot(dens_Pop,num_Phys, xlab = "Densidade Populacional", ylab = "Numero de Medicos")
plot(Pct_65Plus,num_Phys, xlab = "Percentagem Populacao Acima de 65 Anos", ylab = "Numero de Medicos")
plot(Total_Person_Inc,num_Phys, xlab = "Renda Per Capita", ylab = "Numero de Medicos")
```




```r
summary(modelo.ex628[[1]], correlation = TRUE, digits = 4)$correlation
```

```
##                   (Intercept)  Total_Pop  Land_Area Total_Person_Inc
## (Intercept)        1.00000000 -0.1043109 -0.4018189       0.03998723
## Total_Pop         -0.10431087  1.0000000 -0.2963280      -0.98754297
## Land_Area         -0.40181885 -0.2963280  1.0000000       0.27353962
## Total_Person_Inc   0.03998723 -0.9875430  0.2735396       1.00000000
```

```r
summary(modelo.ex628[[2]], correlation = TRUE, digits = 4)$correlation
```

```
##                   (Intercept)    dens_Pop  Pct_65Plus Total_Person_Inc
## (Intercept)        1.00000000 -0.03223599 -0.93166787      -0.18643003
## dens_Pop          -0.03223599  1.00000000 -0.03834988      -0.31708518
## Pct_65Plus        -0.93166787 -0.03834988  1.00000000       0.03370437
## Total_Person_Inc  -0.18643003 -0.31708518  0.03370437       1.00000000
```

Ao observar a matriz dispersão e correlação, podemos ver que a única correlação realmente grande é entre as variáveis “Total Population” e “Total Personal Income”, indicando que o ganho tem um crescimento linear com o total da população. Para o resto das variáveis a correlação é relativamente baixa (sendo negativa para as preditoras Área do terreno e População total) com alta concentração na parte inferior esquerda, o que suporta a ideia descrita no item anterior.

itens c-d

Neste exercício, devemos ajustar um modelo de regressão de múltiplo de primeira ordem

$$Y_i = \sum_{i=0}^{k} X_i \beta_i + \epsilon_i$$

cuja função de regressão será

$$E\{Y_i\} = \sum_{i=0}^{k} X_i \beta_i$$

onde $X_0 = 1$ e $k = 3$. O código que realiza a regressão para os modelo1 e modelo2 propostos seguem abaixo.

```r
modelo1 = lm(num_Phys ~ Total_Pop + Land_Area + Total_Person_Inc)
modelo2 = lm(num_Phys ~ Total_Pop/Land_Area + Pct_65Plus + Total_Person_Inc)
```

Adicionalmente, gostariamos de utilizar o R^2 múltiplo para ter uma palpite inicial de qual seria o melhor modelo ajustado pela reta de regressão de multipla de primeira ordem. Estes valores para cada modelo são dados a seguir. Note que eles são muito próximos logo, para este exemplo, o R^2 múltiplo não nos permite tirar qualquer conclusão.

```r
(summary(modelo1)$r.squared)
```

```
## [1] 0.9026432
```

```r
(summary(modelo2)$r.squared)
```

```
## [1] 0.9040885
```

item e

Para este, basta utilizar "plot(modelo.ex143[[i]], which = c(j,k))", caso outra pessoal, além de mim, que vá resolver nao perca tempo

Exercício 6.29

itens a-b

Para este exercício, usaremos o conjunto de dados CDI para entender como os números de crimes graves, densidade populacional, ganho pessoal per capita e porcentagem de graduados com ensino médio se relacionam em cada região geográfica. Em especial, faremos uma regressão do número de crimes graves contra as outras variáveis, ou seja, tentaremos explicar o comportamento de Y(número de crimes graves) utilizando

X_1(densidade populacional), X_2(ganho pessoal per capita) e X_3(porcentagem de graduados com ensino médio). O modelo proposto para isto será

$$Y_i = \sum_{i=0}^{k} X_i\beta_i + \epsilon_i$$

cuja função de regressão é

$$E\{Y_i\} = \sum_{i=0}^{k} X_i\beta_i$$

onde $X_0 = 1$ e $k = 3$. Para este modelo e, para cada um das 4 regiões geográficas (i=4), a técnica de regressão utilizada é dada pelo código abaixo.

```
modelo1.ex629 = vector("list",max(Geographic_Region))
beta.ex629 = matrix(0,nrow=max(Geographic_Region),ncol=4)
colnames(beta.ex629) = c("beta0","beta1","beta2","beta3")
rownames(beta.ex629) = c("regiao1","regiao2","regiao3","regiao4")
Dens=Total_Pop/Land_Area

for(i in 1:max(Geographic_Region)){
  modelo1.ex629[[i]] = lm(Total_Crimes[which(Geographic_Region ==i)] ~ Dens[which(Geographic_Region==i)]
  beta.ex629[i,] = coef(modelo1.ex629[[i]])
}
(round(beta.ex629,digits = 3))
```

```
##             beta0  beta1 beta2    beta3
## regiao1 -26139.09 16.336 0.383  291.068
## regiao2  63104.12  2.588 3.602 -854.549
## regiao3  56929.39  0.306 4.896 -800.396
## regiao4  37724.58 -0.992 3.627 -489.015
```

Portanto, as funções estimadas $\hat{Y_i}$ são

1. $\hat{Y_{1i}}$ = -26139.09 + 16.336X_{1i} + 0.383X_{2i} + 291.068X_{3i}
2. $\hat{Y_{2i}}$ = 63104.12 + 2.588X_{1i} + 3.602X_{2i} - 854.549X_{3i}
3. $\hat{Y_{3i}}$ = 56929.39 + 0.306X_{1i} + 4.896X_{2i} - 800.396X_{3i}
4. $\hat{Y_{4i}}$ = 37724.58 - 0.992X_{1i} + 3.627X_{2i} - 489.015X_{3i}

Logo, podemos concluir que as únicas funções parecidas nos parâmetros são as funções 2 e 3.

item c

Os valores dos R^2 e MSE para cada região geográfica são dados a seguir.

```
r2 = rep(0,times=max(Geographic_Region))
MSE.ex629 = rep(0,times=max(Geographic_Region))

for(i in 1:max(Geographic_Region)){
  r2[i] = summary(modelo1.ex629[[i]])$r.squared
  MSE.ex629[i] = summary(modelo1.ex629[[i]])$sigma^2
}
```

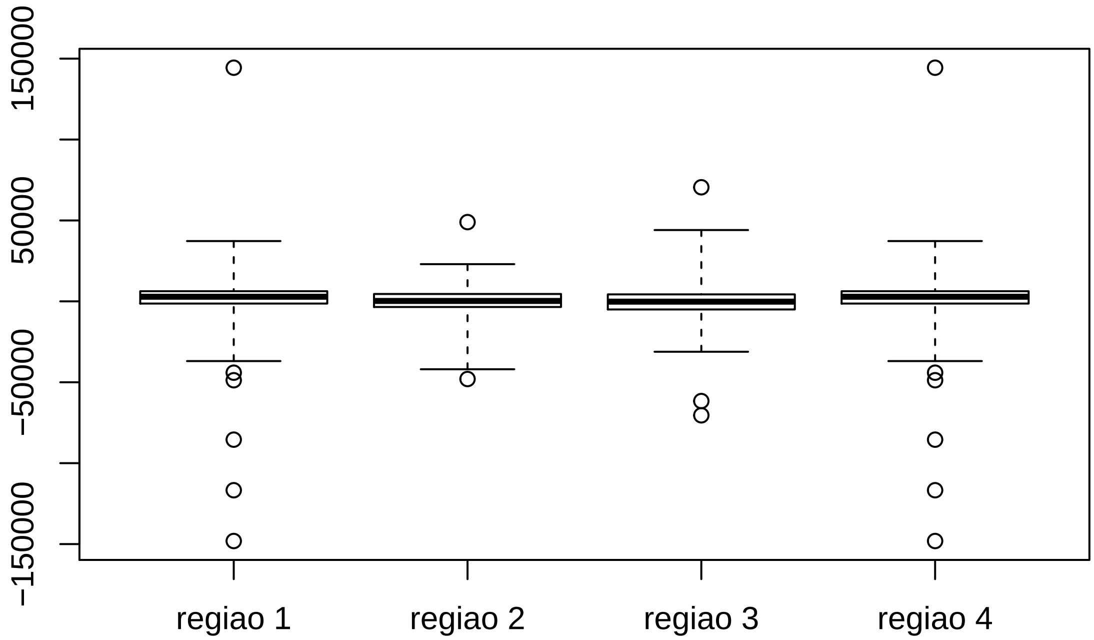
```
##                 regiao1      regiao2      regiao3      regiao4
## r2                 0.83         0.94         0.87         0.97
## MSE.ex629 807279589.89 140169093.43 197074515.96 210421819.84
```

Ao ver os resultados de R^2 e MSE por região podemos ver que os valores de R^2 estão altos mas não tão próximos, ao passo que, para os MSE, podemos notar uma grande variância. Portanto, podemos concluir que as similaridades entre regiões, para estes aspectos, só são preservadas para algumas regiões específicas nos valores de R^2.

item d

Ao fazermos os boxplots dos resíduos de cada região, podemos notar que

```
boxplot(modelo1.ex629[[1]]$residuals,modelo1.ex629[[2]]$residuals,modelo1.ex629[[3]]$residuals,modelo1.e
```



Para todas as regiões, podemos ver que há uma grande concentração de massa em torno do zero. Além disso, ao observarmos os valores máximos e mínimos de cada gráfico, para a escala atual, percebemos que a dispersão do resíduo é relativamente similar entre as regiões. Por fim, o ponto mais claro é a questão dos outliers, sendo bem visível para os boxplots 1 e 4 (os casos 2 e 3 são discutíveis).