Que transformações são possíveis para atingir a normalidade dos dados? Realize uma pesquisa sobre este assunto, apresente um exemplo completo.

5 Aula 5

5.1 Dados não métricos com variáveis dicotômicas

Variável Sexo, possui 2 categorias feminino ou masculino. Definimos uma variável dicotômica X1 = 1 se feminino ocorre e 0 em caso contrário. Ou ainda X1 = 1 se masculino ocorre e 0 em caso contrário. A categoria omitida refere-se ao grupo de comparação. Assim na modelagem se X1 = 1 for para o caso feminino, estaremos comparando o resultado feminino em relação a categoria omitida que é o grupo masculino. (Hair pg 86 e 87 tabela 2.13 e 2.14)

5.2 Multicolinearidade

Ocorre quando qualquer variável independente é altamente correlacionada com o conjunto de outra variáveis independentes.

O ideal é ter diversas variáveis independentes altamente correlacionadas com a variável dependente, mas com pouca correlação entre elas próprias

5.3 Os efeitos da multicolinearidade

O caso extremo de colinearidade ou multicolinearidade no qual uma variável independente é perfeitamente prevista (uma correlação de ± 1,0) por uma ou mais variáveis independentes.Modelos de regressão não podem ser estimados quando existe uma singularidade. O pesquisador deve omitir uma ou mais das variáveis independentes envolvidas para remover a singularidade.

Um exemplo simples:

```
M=matrix(c(1,2,4,2,4,8,3,6,12,5,10,1),ncol=3,nrow=4, byrow = T)

M
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    4
## [2,]    2    4    8
## [3,]    3    6   12
## [4,]    5   10    1
```

```
cor(M)
```

```
##            [,1]       [,2]       [,3]
## [1,]  1.0000000  1.0000000 -0.3159813
## [2,]  1.0000000  1.0000000 -0.3159813
## [3,] -0.3159813 -0.3159813  1.0000000
```

```
lm(M[,3]~M[,1]+M[,2]) #singularidade não permite estimar o parâmetro
```

```
##
## Call:
## lm(formula = M[, 3] ~ M[, 1] + M[, 2])
##
## Coefficients:
## (Intercept)       M[, 1]       M[, 2]
##      8.6857      -0.8857           NA
```

```
lm(M[,1]~M[,2]+M[,3])
```

```
##
## Call:
## lm(formula = M[, 1] ~ M[, 2] + M[, 3])
##
## Coefficients:
## (Intercept)       M[, 2]       M[, 3]
##   4.441e-16    5.000e-01    1.411e-17
```

Um exemplo com 4 variáveis simulando um conjunto de dados com uma estrutura de correlação variada. Mais detalhes pode ser visto aqui

```
require(MASS)
require(clusterGeneration)

set.seed(20)
num.vars<-4
num.obs<-30
cov.mat<-genPositiveDefMat(num.vars,covMethod="unifcorrmat")$Sigma
X<-mvrnorm(num.obs,rep(0,num.vars),Sigma=cov.mat)
print(cor(X), digits = 1)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  1.0  0.6 -0.5  0.9
## [2,]  0.6  1.0 -0.5  0.5
## [3,] -0.5 -0.5  1.0 -0.7
## [4,]  0.9  0.5 -0.7  1.0
```

Agora vamos simular a variável resposta y como combinação linear das quatro variáveis geradas anteriormente mais um erro aleatório.

```
set.seed(2)
parms<-runif(num.vars,-10,10)
y<-X %*% matrix(parms) + rnorm(num.obs,sd=2)
```

Agora vamos ajustar um modelo de regressão: y ~ x1+x2+x3+x4

```
dados<-data.frame(y,X)
form.in<-paste('y ~',paste(names(dados)[-1],collapse='+'))
form.in
```

```
## [1] "y ~ X1+X2+X3+X4"
```

```
mod1<-lm(y ~ X1+X2+X3+X4,data=dados)
summary(mod1)
```

```
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4, data = dados)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7476 -0.8732  0.2273  1.5432  3.2699
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.4505     0.4573   0.985    0.334
```

```
## X1          -4.3579     4.1721  -1.045    0.306
## X2           3.7716     0.7342   5.137 2.61e-05 ***
## X3           0.6487     1.4254   0.455    0.653
## X4          -9.0009     5.3571  -1.680    0.105
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.416 on 25 degrees of freedom
## Multiple R-squared:  0.9786, Adjusted R-squared:  0.9751
## F-statistic: 285.2 on 4 and 25 DF,  p-value: < 2.2e-16
```

Esperaríamos que um modelo de regressão indicasse que cada uma das quatro variáveis explanatórias estão significativamente relacionadas à variável resposta y, uma vez que sabemos a verdadeira relação de y com cada uma das variáveis. No entanto, devemos lembrar que nossas variáveis explicativas estão correlacionadas. O que acontece quando criamos o modelo?

Observamos que apenas a variável X_2 esta significativamente relacionada à variável resposta (com $\alpha = 0,05$), mas sabemos que cada uma das variáveis está relacionada a y.

Podemos tentar uma abordagem alternativa para construir o modelo que considera a colinearidade entre as variáveis explicativas, ou seja, precisamos avaliar a multicolinearidade.

5.4 Avaliação da multicolinearidade

Hair (pag 190 6a. edicao) Uma questão-chave na interpretação da variável estatística de regressão é a correlação entre as variáveis independentes. Esse é um problema de dados, e não de especificação de modelo. A situação ideal para um pesquisador seria ter diversas variáveis independentes altamente correlacionadas com a variável dependente, mas com pouca correlação entre elas próprias.

A tarefa do pesquisador inclui o seguinte:

- Avaliar o grau de multicolinearidade.
- Determinar seu impacto sobre os resultados.
- Aplicar as necessárias ações corretivas, se for o caso.

A maneira mais simples e óbvia de identificar colinearidade é um exame da matriz de correlação para as variáveis independentes. A presença de elevadas correlações (geralmente 0,90 ou maiores) é a primeira indicação de colinearidade substancial. No entanto, a falta de valores elevados de correlação não garante ausência de colinearidade.

Colinearidade pode ser proveniente do efeito combinado de duas ou mais variáveis independentes (o que se chama de multicolinearidade). Para avaliar multicolinearidade precisamos de uma medida que expresse o grau em que cada variável independente é explicada pelo conjunto de outras variáveis independentes. Em termos simples, cada variável independente se torna uma variável dependente e é regredida relativamente às demais variáveis independentes. As duas medidas mais comuns para se avaliar colinearidade aos pares ou múltipla são a tolerância e sua inversa, o fator de inflação de variância conhecido como VIF (Variance Inflation Factor).

5.5 Como detectar a multicolinearidade?

Alguns métodos serão listados abaixo:

1. **Coeficiente de correlação simples**: é uma medida comumente usada no caso de duas variáveis independentes, sendo suficiente para detectar a colinearidade. Considera-se que um coeficiente de correlação maior que 0,80 ou 0,90 é indicativo de colinearidade. Porém, para mais de duas variáveis independentes, mesmo os coeficientes de correlação sendo baixos ainda pode existir a multicolinearidade, pois pares de correlações podem não dar visão de intercorrelacionamentos mais complexos entre três ou mais variáveis;

```
#Observa-se alta correlação entre V1 e V4

print(cor(X), digits = 1)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  1.0  0.6 -0.5  0.9
## [2,]  0.6  1.0 -0.5  0.5
## [3,] -0.5 -0.5  1.0 -0.7
## [4,]  0.9  0.5 -0.7  1.0
```

2. **Determinante da matriz de correlação**: analisar o determinante da matriz de correlações entre as variáveis independentes. Um valor deste determinante próximo de zero é indicativo de multicolinearidade.

```
#Observa-se determinante próximo de zero.
det(cor(X))
```

```
## [1] 0.001864246
```

3. **Autovalores**: Sejam λ_i , i=1,...,p, os autovalores da matriz de correlação das variáveis independentes. Obtenha L, dado por $L = \lambda_{máx}/\lambda_{mín}$, onde $\lambda_{máx}$ é o maior autovalor e $\lambda_{mín}$ é o menor autovalor. Se $L < 100$, considera-se não existir multicolinearidade, se $100 \leq L \leq 1000$ existe multicolinearidade moderada e se $L > 1000$ há indicativo de forte multicolinearidade.

```
#Observa-se L entre 100 e 1000 o que indica multicolinearidade moderada.

lambda=eigen(cor(X))$values
```

```
L= max(lambda)/min(lambda)
L
```

```
## [1] 1311.365
```

4. **Tolerância/VIF**: uma forma de descobrir qual variável X_i está relacionada a outras variáveis independentes X_1, X_2, ..., X_n é fazer a regressão de cada X_i contra as demais variáveis X e calcular o R^2 correspondente (R_i^2). A tolerância é dada por $1 - R_i^2$ e o $VIF_i = \frac{1}{1-R_i^2}$. Se R_i^2 aumenta no sentido da unidade, a colinearidade de X_i com os outros regressores também aumenta. Então o VIF também aumenta e, no limite tende a infinito.

```
#A interpretação de VIF poderá ser vista na tabela a seguir.

car::vif(mod1)
```

```
##        X1        X2        X3        X4
## 175.78787  15.57748  36.66688 224.17152
```

5.6 Tolerância/VIF

Para qualquer modelo de regressão com duas ou mais variáveis independentes, a tolerância pode ser simplesmente definida em dois passos:

1. Considere cada variável independente, uma por vez, e calcule R^2 (coeficiente de variação entre a variável em questão e todas as demais variáveis independentes no modelo de regressão). Neste processo, a variável independente escolhida é transformada em uma dependente prevista pelas demais.

2. Tolerância é então calculada como 1 – R^2 e o VIF = $\frac{1}{1-R^2}$. Por exemplo, se as outras variáveis independentes explicam 25% da variável independente X_1 (R^2 = 0,25), então o valor de tolerância de X_1 é 0,75 (1,0 – 0,25 = 0,75) e o VIF é 1.33.

O valor de tolerância deve ser alto, o que significa um pequeno grau de multicolinearidade (i.e., as outras variáveis independentes coletivamente não têm qualquer quantia considerável de variância compartilhada).

Abaixo, tabela que indica a relação entre o aumento do grau de correlação entre as variáveis e o aumento do VIF, ou seja, quanto maior a correlação entre as variáveis dependentes maior será o VIF e o nível dessa correlação:

	R2	Tol	VIF	Niveis
até 0.4	0.40	0.60	1.67	Fraca
próximo de 0.6	0.60	0.40	2.50	Média
próximo de 0.75	0.75	0.25	4.00	Forte
próximo de 0.85	0.85	0.15	6.67	Muito Forte
0.9 ou mais	0.90	0.10	10.00	Fortíssima

Tem pacote do R para calcular o VIF? Tem vários. Um deles é o **car**.

Voltando ao nosso modelo:

```
summary(mod1)
```

```
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4, data = dados)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7476 -0.8732  0.2273  1.5432  3.2699
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.4505     0.4573   0.985    0.334
## X1           -4.3579     4.1721  -1.045    0.306
## X2            3.7716     0.7342   5.137 2.61e-05 ***
## X3            0.6487     1.4254   0.455    0.653
## X4           -9.0009     5.3571  -1.680    0.105
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.416 on 25 degrees of freedom
## Multiple R-squared:  0.9786, Adjusted R-squared:  0.9751
## F-statistic: 285.2 on 4 and 25 DF,  p-value: < 2.2e-16
```

```
car::vif(mod1)
```

```
##        X1        X2        X3        X4
## 175.78787  15.57748  36.66688 224.17152
```

A variável X4 é a que apresenta o maior valor para VIF. Vamos remover esta variável e repetir a análise.

```
mod2<-lm(y ~ X1+X2+X3,data=dados)
summary(mod2)
```

```
##
## Call:
## lm(formula = y ~ X1 + X2 + X3, data = dados)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3324 -1.2362 -0.0395  1.8609  4.1507
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.5305     0.4705   1.128     0.27
## X1          -11.3328     0.4314 -26.267  < 2e-16 ***
## X2            4.9347     0.2529  19.510  < 2e-16 ***
## X3            2.9938     0.2994   9.999 2.13e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.499 on 26 degrees of freedom
## Multiple R-squared:  0.9761, Adjusted R-squared:  0.9734
## F-statistic: 354.5 on 3 and 26 DF,  p-value: < 2.2e-16
```

```
car::vif(mod2)
```

```
##       X1       X2       X3
## 1.756630 1.727845 1.511878
```

O modelo 2 de regressão é muito melhor que o modelo 1. Observamos um ajuste melhor do número de variáveis que estão significativamente relacionadas à variável resposta.

5.7 Exercício:

Pegue uma base de dados (Iris, attitude, Orange) e realize a análise de dados discrepantes, multicolinearidade e teste de normalidade.

```
#Dica: se for necessário, faça uma seleção de variáveis.
```

2- Crie uma matriz de dados multivariada com 10 variáveis explicativas, 1 variável resposta e 200 observações. Teste a multicolinearidade e proponha um modelo de regressão para y.

```
#Dica: tome como base o último exemplo com a matriz X (linha 356 do arquivo Rmd)
```

6 Avaliação dos conceitos iniciais

Prova escrita.

6.1 Referências

Cheng, X., Cook, D., & Hofmann, H. (2015). Visually Exploring Missing Values in Multivariable Data Using a Graphical User Interface. Journal of Statistical Software, 68(6), 1 - 23. doi:http://dx.doi.org/10.18637/jss.v068.i06

Jamshidian, M., Jalal, S., & Jansen, C. (2014). MissMech: An R Package for Testing Homoscedasticity, Multivariate Normality, and Missing Completely at Random (MCAR). Journal of Statistical Software, 56(6), 1 - 31. doi:http://dx.doi.org/10.18637/jss.v056.i06

https://beckmw.wordpress.com/2013/02/05/collinearity-and-stepwise-vif-selection/

```
print(sessionInfo(), locale = FALSE)
```

```
## R version 3.4.3 (2017-11-30)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 17134)
##
## Matrix products: default
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
```

```
##  [1] kableExtra_0.9.0        clusterGeneration_1.3.4
##  [3] MASS_7.3-50             VIM_4.7.0
##  [5] data.table_1.11.4       colorspace_1.3-2
##  [7] mice_3.1.0              lattice_0.20-35
##  [9] MissMech_1.0.2          Amelia_1.7.5
## [11] Rcpp_0.12.17            DescTools_0.99.23
## [13] forcats_0.2.0           stringr_1.3.1
## [15] dplyr_0.7.5             purrr_0.2.4
## [17] readr_1.1.1             tidyr_0.8.1
## [19] tibble_1.4.2            ggplot2_2.2.1
## [21] tidyverse_1.2.1
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-131       lubridate_1.7.4   httr_1.3.1
##  [4] rprojroot_1.3-2    tools_3.4.3       backports_1.1.2
##  [7] utf8_1.1.3         R6_2.2.2          rpart_4.1-11
## [10] lazyeval_0.2.1     jomo_2.6-2        nnet_7.3-12
## [13] sp_1.3-1           tidyselect_0.2.3  mnormt_1.5-5
## [16] curl_3.1           compiler_3.4.3    cli_1.0.0
## [19] rvest_0.3.2        expm_0.999-2      xml2_1.2.0
## [22] labeling_0.3       scales_0.5.0      lmtest_0.9-36
## [25] DEoptimR_1.0-8     mvtnorm_1.0-7     psych_1.8.4
## [28] robustbase_0.93-0 digest_0.6.15     foreign_0.8-69
## [31] minqa_1.2.4        rmarkdown_1.10    rio_0.5.9
## [34] pkgconfig_2.0.1    htmltools_0.3.6   lme4_1.1-15
## [37] manipulate_1.0.1   highr_0.6         rlang_0.2.1
## [40] readxl_1.0.0       rstudioapi_0.7    prettydoc_0.2.1
## [43] bindr_0.1.1        zoo_1.8-1         jsonlite_1.5
## [46] car_3.0-0          magrittr_1.5      Matrix_1.2-14
## [49] munsell_0.4.3      abind_1.4-5       stringi_1.1.7
## [52] yaml_2.1.18        carData_3.0-0     plyr_1.8.4
## [55] parallel_3.4.3     mitml_0.3-5       crayon_1.3.4
## [58] haven_1.1.1        splines_3.4.3     hms_0.4.1
## [61] knitr_1.20         pillar_1.1.0      boot_1.3-20
## [64] mvnormtest_0.1-9   reshape2_1.4.3    pan_1.6
## [67] glue_1.2.0         evaluate_0.10.1   laeken_0.4.6
## [70] modelr_0.1.1       vcd_1.4-4         nloptr_1.0.4
## [73] cellranger_1.1.0   gtable_0.2.0      assertthat_0.2.0
## [76] openxlsx_4.0.17    broom_0.4.3       e1071_1.6-8
## [79] viridisLite_0.3.0 class_7.3-14      survival_2.41-3
## [82] bindrcpp_0.2.2
```