3º Estágio / Trabalho Avaliativo

Ítallo Silva - 118110718 | Thiago Nascimento - 118110804 | João Marcelo Junior - 117110448

Questão 3

Carregando os dados

A seguir, fazemos a leitura dos dados, neles encontramos três colunas:

- faturamento, nossa variável *target*;
- pesquisa
- propaganda

```
dados <- read_xls("../datasets/faturamento-hamburguer-fast-food.xls")
```

Separação em treino e teste

Iremos separar nosso conjunto de dados em treino e teste, com as proporções de 80% e 20%, respectivamente. Com isso, poderemos avaliar o desempenho do modelo em dados que este não viu previamente descartando a possibilidade de *overfitting*.

```
set.seed(1223)
treino <- sample_frac(dados, .8)
teste <- setdiff(dados, treino)
```

Criando o modelo

Criaremos o modelo linear utilizando a função `lm`.

```
mod <- lm(faturamento ~ ., treino)
```

Testando os pressupostos

Normalidade dos resíduos

Para avaliar a normalidade dos resíduos, aplicaremos o teste de Shapiro-Wilk, nesse teste temos:

- H_0 : distribuição dos dados = normal
- H_1 : distribuição dos dados $\neq$ normal

```r
shapiro.test(mod$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  mod$residuals
## W = 0,95, p-value = 0,08
```

Temos um p-value de aproximadamente 0.08, assim considerando um nível de significância de 5% podemos assumir a normalidade dos resíduos.

Outliers nos resíduos

```r
summary(rstandard(mod))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -1,9317 -0,8277  0,0382 -0,0021  1,0014  1,6370
```

Vemos que a distribuição dos resíduos padronizados está entre -2 e 2, assim podemos concluir que não temos nenhum outlier.

Independência dos resíduos

Iremos verificar a independência dos resíduos a partir do teste Durbin-Watson, nele temos as seguintes hipóteses:

- $H_0 : \rho = 0$, não há autocorrelação dos resíduos
- $H_1 : \rho \neq 0$, há autocorrelação dos resíduos

```r
car::durbinWatsonTest(mod)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1         -0,1565         2,282   0,316
##  Alternative hypothesis: rho != 0
```

Considerando um nível de significância de 5%, como o p-value é maior que ele, assumimos a hipótese nula, assim confirmando que os resíduos são independentes.

Homocedasticidade

Verificaremos a seguir a presença de homocedasticidade, ou seja, variância constante dos resíduos, temos como hipóteses:

- H_0 : há homocedasticidade
- H_1 : não há homocedasticidade

```r
lmtest::bptest(mod)
```

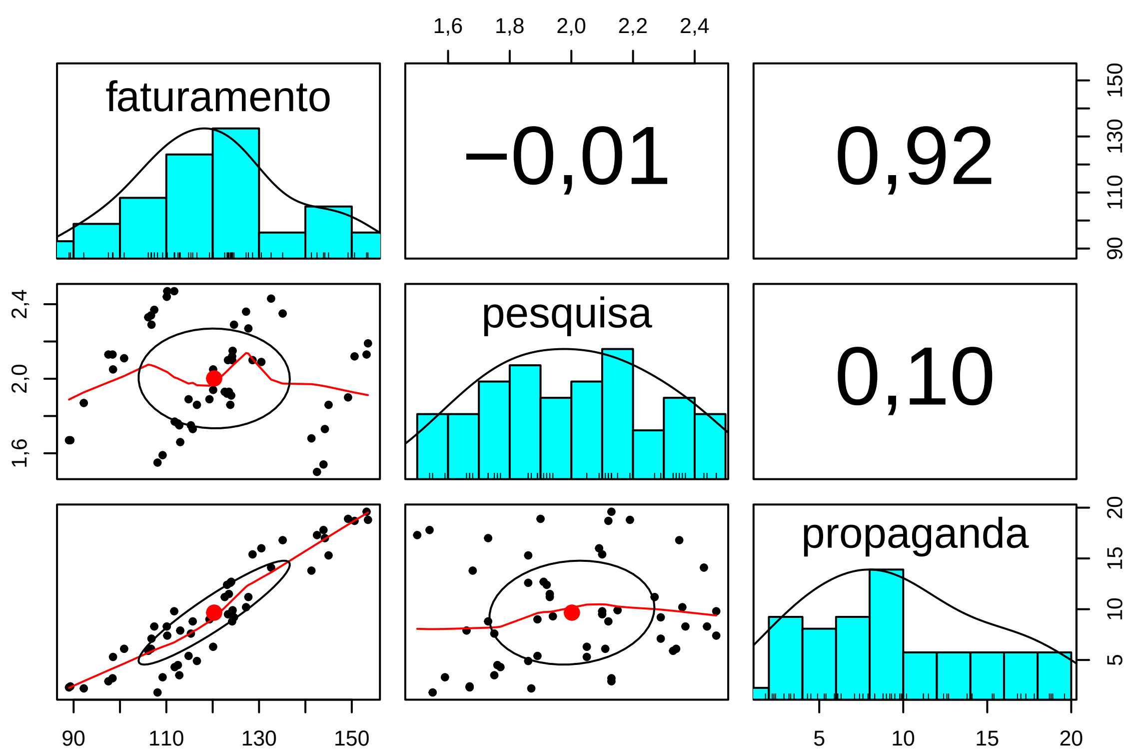
```
##
##  studentized Breusch-Pagan test
##
## data:  mod
## BP = 7,9, df = 2, p-value = 0,02
```

Temos um p-value menor que 5%, sendo assim devemos assumir a hipótese alternativa e portanto não temos homocedasticidade.

Ausência de multicolinearidade

Vamos então avaliar se existe multicolinearidade, ou seja, alguma correlação muito alta entre as variáveis independentes. Consideraremos uma correlação alta se ela for maior que 0.8, em módulo.

```r
psych::pairs.panels(dados)
```



Pelo gráfico acima, vemos que a correlação entre as variáveis independentes é bastante baixa (0.10), assim indicando que não há multicolinearidade. Podemos ver ainda que a variável pesquisa tem uma baixa correlação com a variável alvo faturamente, sugerindo que talvez ela não seja necessária ao modelo.

Outra forma de verificar a ausência de multicolinearidade é através da função VIF (inflação da variância):

```r
car::vif(mod)
```

```
##   pesquisa propaganda
##      1,022      1,022
```

Consideramos que existe multicolinearidade caso o valor de VIF da variável seja maior que 10, como vemos não é nosso caso. Assim, confirmamos que não temos multicolinearidade.

Analisando o modelo

```r
summary(mod)
```

```
##
## Call:
## lm(formula = faturamento ~ ., data = treino)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11,280  -4,846   0,225   5,929   9,864
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  102,195      7,155   14,28   <2e-16 ***
## pesquisa      -5,263      3,586   -1,47     0,15
## propaganda     2,989      0,181   16,47   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0,001 '**' 0,01 '*' 0,05 '.' 0,1 ' ' 1
##
## Residual standard error: 6,14 on 39 degrees of freedom
## Multiple R-squared:  0,875,  Adjusted R-squared:  0,868
## F-statistic:  136 on 2 and 39 DF,  p-value: <2e-16
```

Pelo exposto acima, vemos que a variável pesquisa apresenta uma baixa significância no modelo (o coeficiente da variável é próximo a zero), conforme suposto pela sua baixa correlação.

Podemos ver ainda que obtivemos um R^2 ajustado de 0.86, e vemos que a estatística F tem um p-value muito baixo, assim com muita significância podemos afirmar que o modelo é melhor que o modelo nulo (ou seja, simplesmente prever utilizando a média).

Este resultado pode indicar que temos um bom modelo ou que está havendo *overfitting*, precisamos então realizar um teste para avaliar o R^2 do modelo com dados desconhecidos.

Realizando predição com o modelo

Iremos realizar uma predição usando o nosso modelo treino e o conjunto de dados que separamos como conjunto de teste e avaliar o R^2 obtido.

```r
predicoes <- predict.lm(mod, teste)
yardstick::rsq_vec(teste$faturamento, predicoes)
```

```
## [1] 0,807
```

Obtivemos um R^2 de aproximadamente 0.81, indicando que temos de fato um bom modelo.

Questão 4

Carregando os dados

Os dados provém , os mesmos se referem aos preços de casas nos EUA.

A seguir, fazemos a leitura dos dados, neles encontramos três colunas:

- Ganho médio por área
- Idade da casa
- Número de cômodos
- Número de quartos
- População da área
- Preço

```
dados <- read_csv("../datasets/House_price.csv", col_types = cols_only(
  `Avg. Area Income` = col_double(),
  `House Age` = col_double(),
  `Number of Rooms` = col_double(),
  `Number of Bedrooms` = col_double(),
  `Area Population` = col_double(),
  Price = col_double()
))
```

Separação em treino e teste

Iremos separar nosso conjunto de dados em treino e teste, com as proporções de 80% e 20%, respectivamente. Com isso, poderemos avaliar o desempenho do modelo em dados que este não viu previamente descartando a possibilidade de *overfitting*.

```
set.seed(1223)
treino <- sample_frac(dados, .8)
teste <- setdiff(dados, treino)
```

Criando o modelo

Criaremos o modelo linear utilizando a função `lm`.

```
mod <- lm(Price ~ ., treino)
```

Testando os pressupostos

Normalidade dos resíduos

Para avaliar a normalidade dos resíduos, aplicaremos o teste de Shapiro-Wilk, nesse teste temos:

- H_0 : distribuição dos dados = normal
- H_1 : distribuição dos dados $\neq$ normal

```r
shapiro.test(mod$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  mod$residuals
## W = 1, p-value = 0,3
```

Temos um p-value de aproximadamente 0,35, sendo então o p-value de grande valor, podemos assumir a normalidade dos resíduos.

Outliers nos resíduos

```r
summary(rstandard(mod))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -3,352  -0,691  -0,003   0,000   0,687   3,405
```

Vemos que a distribuição dos resíduos padronizados está entre -3,5 e 3,5, assim podemos concluir que não temos nenhum outlier.

Independência dos resíduos

Iremos verificar a independência dos resíduos a partir do teste Durbin-Watson, nele temos as seguintes hipóteses:

- $H_0 : \rho = 0$, não há autocorrelação dos resíduos
- $H_1 : \rho \neq 0$, há autocorrelação dos resíduos

```r
car::durbinWatsonTest(mod)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1         0,01321         1,973   0,422
##  Alternative hypothesis: rho != 0
```

Temos um p-value de aproximadamente 0,42, sendo então o p-value de grande valor, podemos assumir a normalidade dos resíduos.

Homocedasticidade

Verificaremos a seguir a presença de homocedasticidade, ou seja, variância constante dos resíduos, temos como hipóteses:

- H_0 : há homocedasticidade
- H_1 : não há homocedasticidade

```r
lmtest::bptest(mod)
```

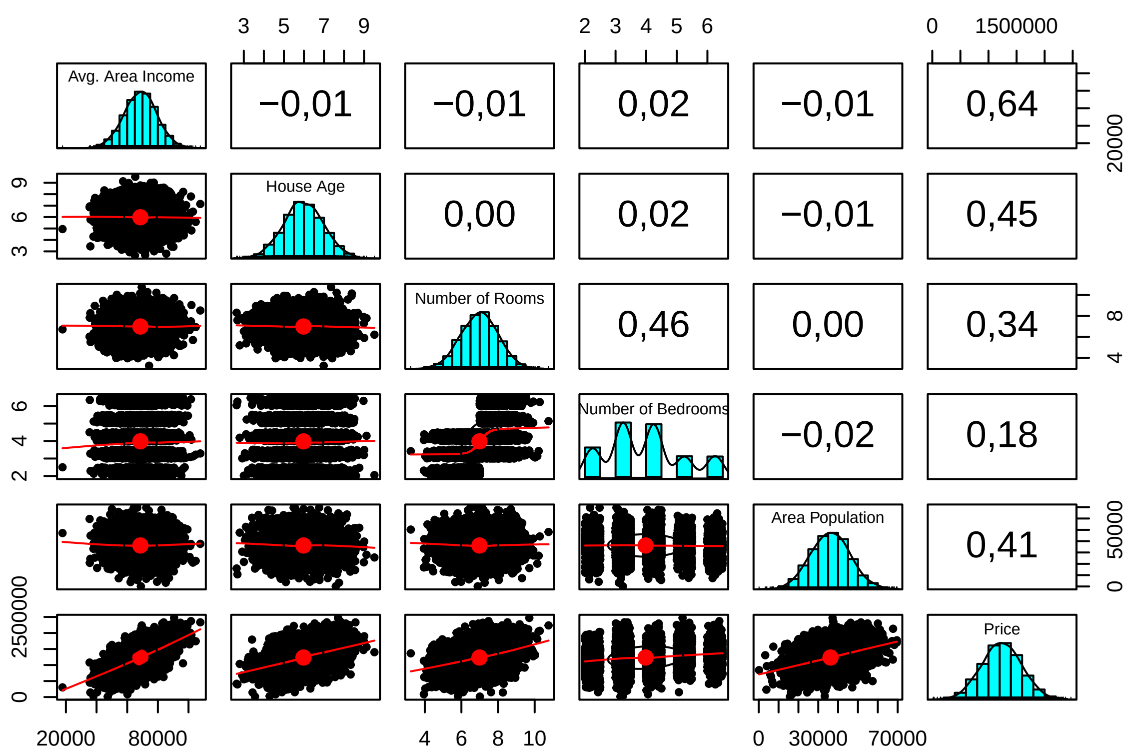
```
##
##  studentized Breusch-Pagan test
##
## data:  mod
## BP = 8,2, df = 5, p-value = 0,1
```

Temos um p-value maior que 5%, sendo assim devemos assumir a hipótese alternativa e portanto temos homocedasticidade.

Ausência de multicolinearidade

Vamos então avaliar se existe multicolinearidade, ou seja, alguma correlação muito alta entre as variáveis independentes. Consideraremos uma correlação alta se ela for maior que 0.8, em módulo.

```r
psych::pairs.panels(dados)
```



Pelo gráfico acima, vemos que a correlação entre as variáveis independentes é bastante baixa, assim indicando que não há multicolinearidade.

Outra forma de verificar a ausência de multicolinearidade é através da função VIF (inflação da variância):

```
car::vif(mod)
```

```
##   `Avg. Area Income`          `House Age`    `Number of Rooms`
##               1,001               1,000               1,269
## `Number of Bedrooms`    `Area Population`
##               1,270               1,001
```

Consideramos que existe multicolinearidade caso o valor de VIF da variável seja maior que 10, como vemos não é nosso caso. Assim, confirmamos que não temos multicolinearidade.

Analisando o modelo

```
summary(mod)
```

```
##
## Call:
## lm(formula = Price ~ ., data = treino)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -340774  -70162    -341   69799  346224
##
## Coefficients:
##                          Estimate   Std. Error t value Pr(>|t|)
## (Intercept)          -2641110,894    20168,434 -130,95   <2e-16 ***
## `Avg. Area Income`         21,586        0,158  136,33   <2e-16 ***
## `House Age`            166888,243     1700,101   98,16   <2e-16 ***
## `Number of Rooms`      121791,800     1899,584   64,11   <2e-16 ***
## `Number of Bedrooms`      202,373     1545,602    0,13      0,9
## `Area Population`          15,063        0,171   87,90   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0,001 '**' 0,01 '*' 0,05 '.' 0,1 ' ' 1
##
## Residual standard error: 102000 on 3632 degrees of freedom
## Multiple R-squared:  0,918,  Adjusted R-squared:  0,918
## F-statistic: 8,1e+03 on 5 and 3632 DF,  p-value: <2e-16
```

Pelo exposto acima, vemos que todas as variáveis tem relevância ao modelo, exceto o número de quartos. Isto se deve ao fato de ter baixa correlação (0,18) à variável alvo.

Podemos ver ainda que obtivemos um R^2 ajustado de 0,92, e vemos que a estatística F tem um p-value muito baixo, assim com muita significância podemos afirmar que o modelo é melhor que o modelo nulo (ou seja, simplesmente prever utilizando a média).

Este resultado pode indicar que temos um bom modelo ou que está havendo *overfitting*, precisamos então realizar um teste para avaliar o R^2 do modelo com dados desconhecidos.

Realizando predição com o modelo

Iremos realizar uma predição usando o nosso modelo treino e o conjunto de dados que separamos como conjunto de teste e avaliar o R^2 obtido.

```r
predicoes <- predict.lm(mod, teste)
yardstick::rsq_vec(teste$Price, predicoes)
```

```
## [1] 0,9227
```

Obtivemos um R^2 de aproximadamente 0.92, indicando que temos de fato um bom modelo.