Inferência Causal em Regressão

Alison Cordeiro Sousa

Este projeto foi estruturado para analisar como fatores como escolaridade, experiência profissional, tempo no emprego atual e gênero influenciam o salário dos trabalhadores. Utilizamos uma base de dados csv contendo informações salariais e características dos indivíduos. Por meio de modelos de regressão linear estimados via mínimos quadrados ordinários (MQO), investigamos relações causais e aplicamos testes t para avaliar a significância individual dos coeficientes, além de testes F para verificar a significância conjunta das variáveis no modelo. Com isso, verificamos se as variáveis consideradas têm efeito estatisticamente relevante no salário. Identificamos que maior escolaridade, mais experiência e maior tempo no emprego estão associados a salários mais altos, enquanto que, mesmo controlando essas variáveis, as mulheres ganham significativamente menos que os homens. Esses resultados evidenciam desigualdades salariais e destacam os principais determinantes do rendimento no mercado de trabalho.

```r
# --- Carregamento das bibliotecas essenciais
library(tidyverse)      # Manipulação, gráficos, etc
library(stargazer)      # Tabelas formatadas
library(car)            # Testes F (linearHypothesis)
library(ggfortify)      # Gráficos diagnósticos

# --- Definição do diretório onde o arquivo wage1.csv está localizado
setwd("C:/Users/PC GAMER/Downloads/data")

# --- Carregamento dos dados
wage1 <- read.csv("wage1.csv")

# --- Criando variável logarítmica para wage para análise de normalidade
wage1$lwage <- log(wage1$wage)

# --- Exibindo as primeiras linhas das variáveis selecionadas
wage1 %>%
  select(wage, educ, exper, tenure, female) %>%
  head(10)
```

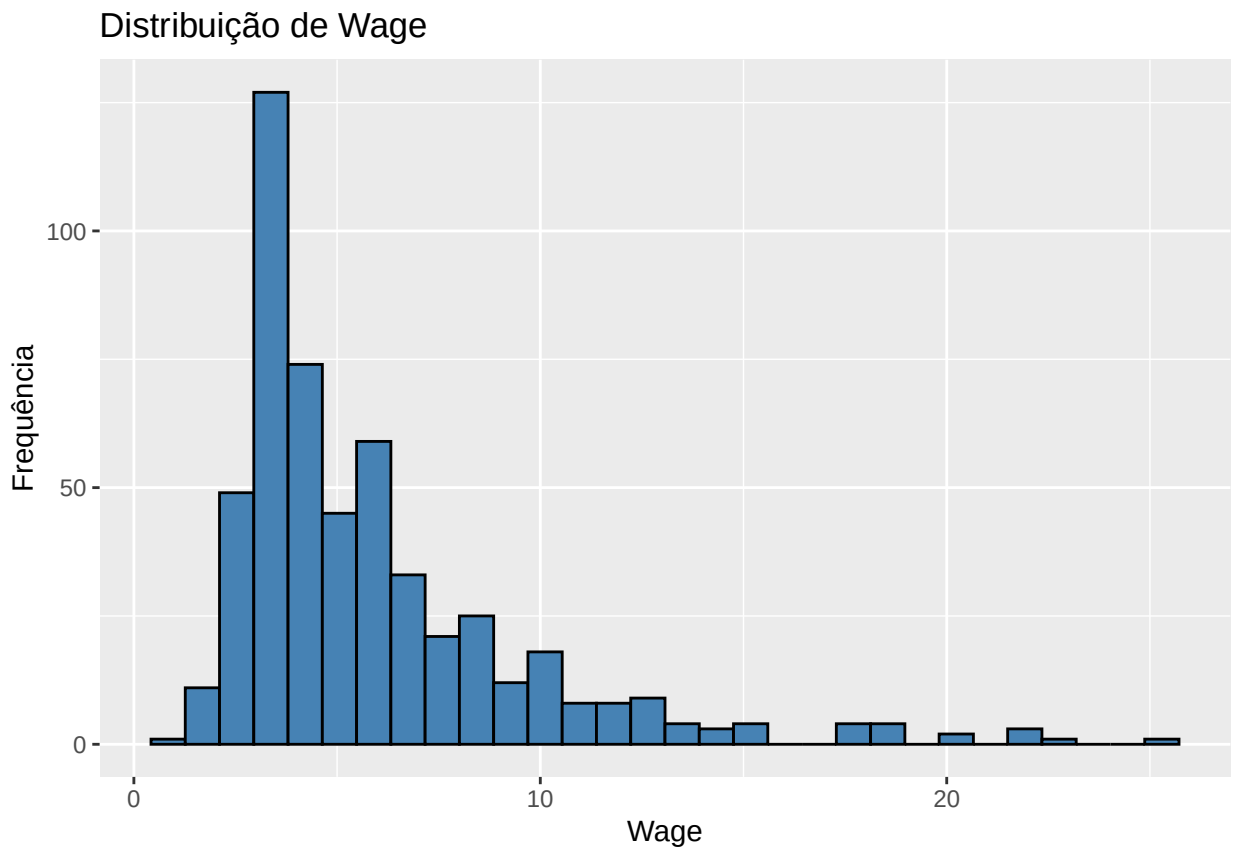
```
##     wage educ exper tenure female
## 1   3.10   11     2      0      1
## 2   3.24   12    22      2      1
## 3   3.00   11     2      0      0
## 4   6.00    8    44     28      0
## 5   5.30   12     7      2      0
```

```
## 6   8.75   16     9      8      0
## 7  11.25   18    15      7      0
## 8   5.00   12     5      3      1
## 9   3.60   12    26      4      1
## 10 18.18   17    22     21      0
```

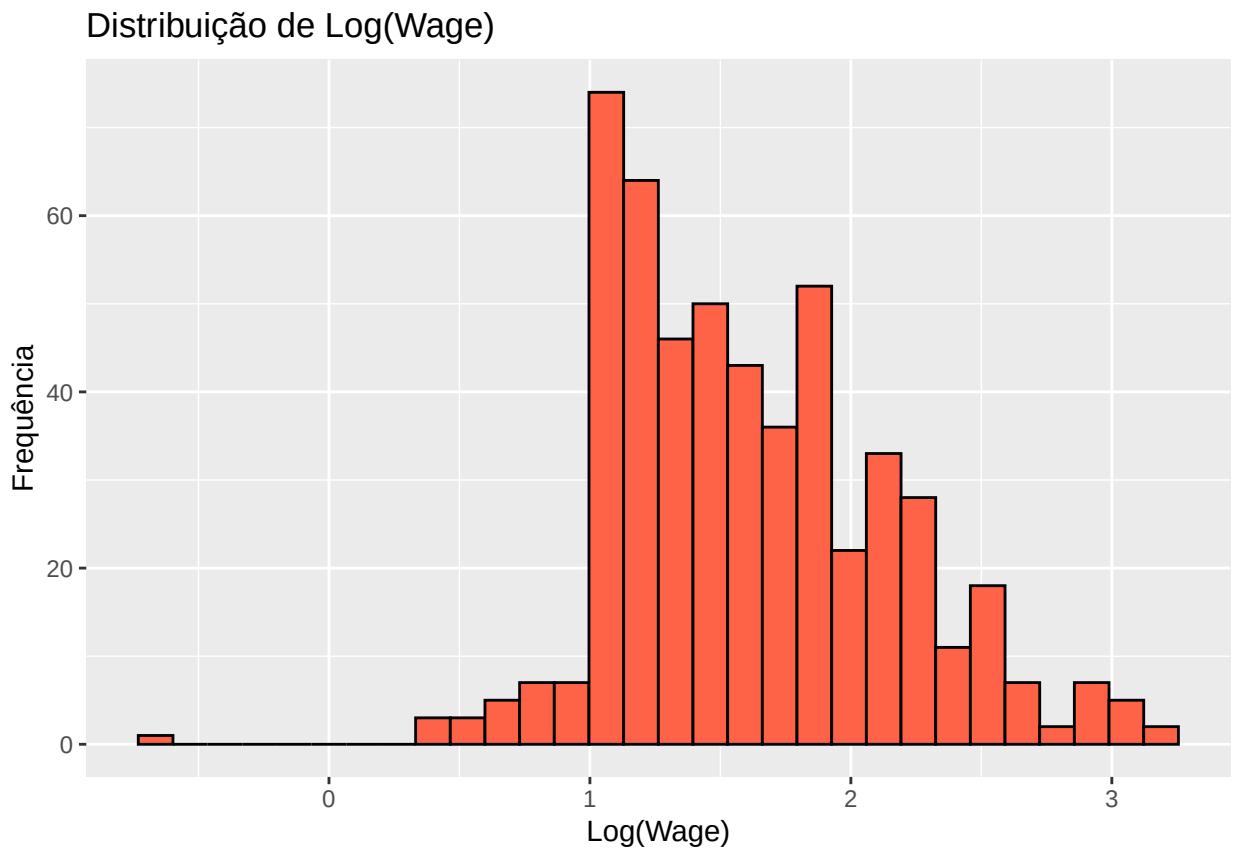
```r
# --- Sumário estatístico detalhado das variáveis,
#     para entender distribuição e características
stargazer(
  wage1 %>% select(wage, educ, exper, tenure, female),
  type = "text",
  title = "Sumário Estatístico Descritivo"
)
```

```
##
## Sumário Estatístico Descritivo
## ========================================
## Statistic  N   Mean  St. Dev.  Min   Max
## ----------------------------------------
## wage      526 5.896   3.693   0.530 24.980
## educ      526 12.563  2.769     0     18
## exper     526 17.017  13.572    1     51
## tenure    526 5.105   7.224     0     44
## female    526 0.479   0.500     0     1
## ----------------------------------------
```

```r
# --- Visualização das distribuições para verificar normalidade

# Histograma da variável wage (salário)
ggplot(wage1) +
  geom_histogram(aes(x = wage), bins = 30, fill = "steelblue", color = "black") +
  labs(
    title = "Distribuição de Wage",
    x = "Wage",
    y = "Frequência"
  )
```




```
# Histograma do log(wage), para verificar se transformação melhora normalidade
ggplot(wage1) +
  geom_histogram(aes(x = lwage), bins = 30, fill = "tomato", color = "black") +
  labs(
    title = "Distribuição de Log(Wage)",
    x = "Log(Wage)",
    y = "Frequência"
  )
```




```
# --- Teste de normalidade Shapiro-Wilk para wage e log(wage)
shapiro_wage <- shapiro.test(wage1$wage)
shapiro_lwage <- shapiro.test(wage1$lwage)
print(shapiro_wage)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  wage1$wage
## W = 0.80273, p-value < 2.2e-16
```

```
print(shapiro_lwage)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  wage1$lwage
## W = 0.96909, p-value = 4.423e-09
```

```r
# --- Estimação do modelo de regressão linear para wage
modelo <- lm(wage ~ educ + exper + tenure + female, data = wage1)

# Sumário completo do modelo, inclui coeficientes, erros padrão, R², etc
summary(modelo)
```

```
##
## Call:
## lm(formula = wage ~ educ + exper + tenure + female, data = wage1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7675 -1.8080 -0.4229  1.0467 14.0075
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.56794    0.72455  -2.164   0.0309 *
## educ         0.57150    0.04934  11.584  < 2e-16 ***
## exper        0.02540    0.01157   2.195   0.0286 *
## tenure       0.14101    0.02116   6.663 6.83e-11 ***
## female      -1.81085    0.26483  -6.838 2.26e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.958 on 521 degrees of freedom
## Multiple R-squared:  0.3635, Adjusted R-squared:  0.3587
## F-statistic:  74.4 on 4 and 521 DF,  p-value: < 2.2e-16
```

```r
# --- Teste t manual para o coeficiente da variável exper

coef_exper <- coef(modelo)["exper"]                    # Coeficiente de exper
se_exper <- sqrt(vcov(modelo)["exper", "exper"])       # Erro padrão de exper
tstat <- coef_exper / se_exper                          # Estatística t calculada
gl_resid <- modelo$df.residual                          # Graus de liberdade residuais

# Valor crítico t bilateral a 5%
tcrit <- qt(0.975, df = gl_resid)

# Valor-p bilateral para teste t do coeficiente de exper
p_valor_t_bilateral <- 2 * pt(abs(tstat), df = gl_resid, lower.tail = FALSE)

# Exibindo resultados do teste t manual (linha a linha, para evitar cortes)
cat("Teste t para coeficiente de exper:\n")
```

```
## Teste t para coeficiente de exper:
```

```r
cat(sprintf("Coeficiente: %.4f\n", coef_exper))
```

```
## Coeficiente: 0.0254
```

```r
cat(sprintf("Erro padrão: %.4f\n", se_exper))
```

```
## Erro padrão: 0.0116
```

```r
cat(sprintf("t-estatística: %.4f\n", tstat))
```

```
## t-estatística: 2.1951
```

```r
cat(sprintf("Valor crítico (5%% bilateral): ±%.4f\n", tcrit))
```

```
## Valor crítico (5% bilateral): ±1.9645
```

```r
cat(sprintf("P-valor bilateral: %.4f\n\n", p_valor_t_bilateral))
```

```
## P-valor bilateral: 0.0286
```

```r
# --- Teste F para significância conjunta das variáveis explicativas

# Modelo restrito (intercepto apenas)
modelo_restrito <- lm(wage ~ 1, data = wage1)

ssr_r <- sum(resid(modelo_restrito)^2)  # Soma dos quadrados dos resíduos do restrito
ssr_ur <- sum(resid(modelo)^2)           # Soma dos quadrados dos resíduos do irrestrito
q <- 4                                   # Número de restrições (coeficientes)

# Estatística F manual
F_stat <- ((ssr_r - ssr_ur) / q) / (ssr_ur / gl_resid)
F_crit <- qf(0.95, df1 = q, df2 = gl_resid)

# Valor-p do teste F manual
p_valor_F <- pf(F_stat, df1 = q, df2 = gl_resid, lower.tail = FALSE)

cat("Teste F para significância conjunta:\n")
```

```
## Teste F para significância conjunta:
```

```r
cat(sprintf("F-Estatística: %.4f\n", F_stat))
```

```
## F-Estatística: 74.3980
```

```r
cat(sprintf("Valor crítico (5%%): %.4f\n", F_crit))
```

```
## Valor crítico (5%): 2.3890
```

```r
cat(sprintf("P-valor: %.4f\n\n", p_valor_F))
```

```
## P-valor: 0.0000
```

```r
# Verificação rápida com car::linearHypothesis
cat("Teste F usando linearHypothesis:\n")
```

```
## Teste F usando linearHypothesis:
```

```r
print(linearHypothesis(modelo, c("educ = 0", "exper = 0", "tenure = 0", "female = 0")))
```

```
##
## Linear hypothesis test:
## educ = 0
## exper = 0
## tenure = 0
## female = 0
##
## Model 1: restricted model
## Model 2: wage ~ educ + exper + tenure + female
##
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    525 7160.4
## 2    521 4557.3  4    2603.1 74.398 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# --- Teste LM (Lagrange Multiplier) para joint significance de exper e tenure

# Modelo restrito (sem exper e tenure)
modelo_r2 <- lm(wage ~ educ + female, data = wage1)

# Obtendo resíduos do modelo restrito
wage1$ehat <- residuals(modelo_r2)
```

```r
# Regressão dos resíduos contra todas as variáveis originais
modelo_ehat <- lm(ehat ~ educ + exper + tenure + female, data = wage1)

R2_ehat <- summary(modelo_ehat)$r.squared
n <- nobs(modelo_ehat)
LM_stat <- n * R2_ehat
q_lm <- 2

# Valor crítico qui-quadrado a 5%
chi2_crit <- qchisq(0.95, df = q_lm)

# Valor-p do teste LM
p_valor_chi2 <- pchisq(LM_stat, df = q_lm, lower.tail = FALSE)

cat("Teste LM para coeficientes de exper e tenure:\n")
```

```
## Teste LM para coeficientes de exper e tenure:
```

```r
cat(sprintf("LM Estatística: %.4f\n", LM_stat))
```

```
## LM Estatística: 74.3190
```

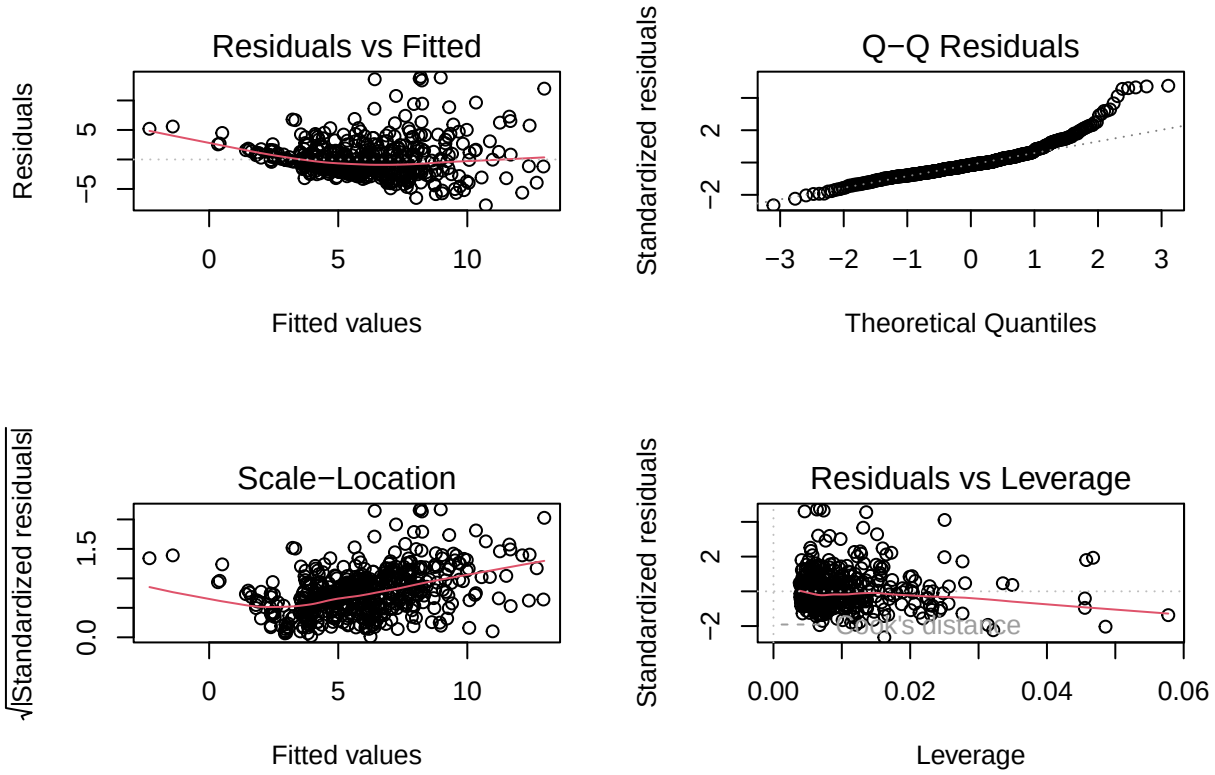
```r
cat(sprintf("Valor crítico Qui-quadrado (5%%): %.4f\n", chi2_crit))
```

```
## Valor crítico Qui-quadrado (5%): 5.9915
```

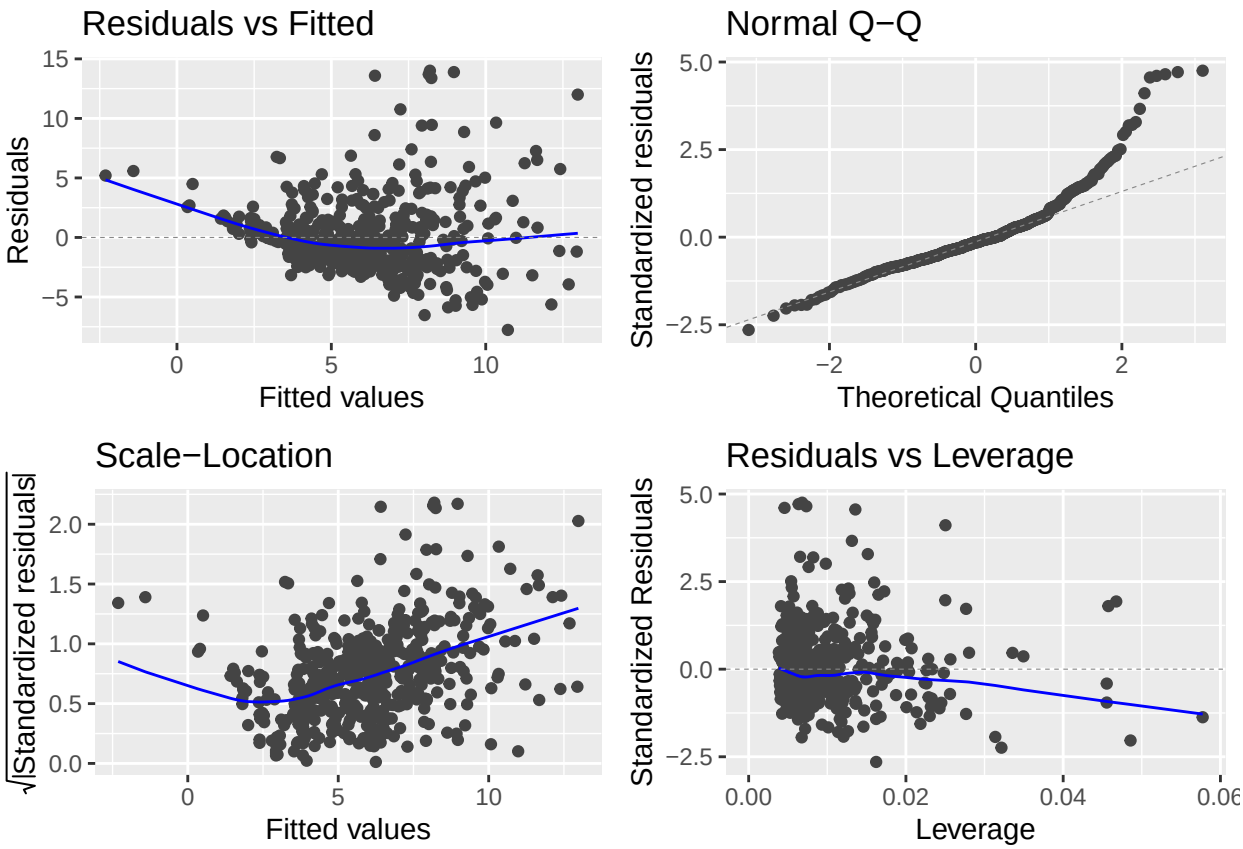
```r
cat(sprintf("P-valor: %.4f\n\n", p_valor_chi2))
```

```
## P-valor: 0.0000
```

```r
# --- Gráficos diagnósticos para validação do modelo e inferência causal

# Configura o painel 2x2
par(mfrow = c(2, 2))

# 1) Residuals vs Fitted sem labels nos pontos
plot(modelo, which = 1, id.n = 0)

# 2) Normal Q-Q sem labels
plot(modelo, which = 2, id.n = 0)

# 3) Scale-Location (Spread-Location) sem labels
plot(modelo, which = 3, id.n = 0)

# 4) Residuals vs Leverage sem labels
plot(modelo, which = 5, id.n = 0)  # Note: 4 é Cook's Distance, 5 é Residuals vs Leverage
```




```
# Restaurar configuração padrão da tela gráfica
par(mfrow = c(1, 1))

# Gráficos aprimorados com ggfortify, sem labels (label.size = 0)
autoplot(modelo, label.size = 0)
```




```
# --- Conclusão organizada e com quebras para evitar cortes

cat("Resumo dos principais resultados inferenciais:\n\n")
```

```
## Resumo dos principais resultados inferenciais:
```

```
cat(sprintf("Coeficiente 'exper': %.4f\n", coef_exper))
```

```
## Coeficiente 'exper': 0.0254
```

```
cat(sprintf("t-Estatística: %.3f\n", tstat))
```

```
## t-Estatística: 2.195
```

```
cat(sprintf("p-valor bilateral: %.4f\n\n", p_valor_t_bilateral))
```

```
## p-valor bilateral: 0.0286
```

```r
cat("Teste F global:\n")
```

```
## Teste F global:
```

```r
cat(sprintf("F = %.3f\n", F_stat))
```

```
## F = 74.398
```

```r
cat(sprintf("p-valor = %.4f\n\n", p_valor_F))
```

```
## p-valor = 0.0000
```

```r
cat("Teste LM para exper e tenure:\n")
```

```
## Teste LM para exper e tenure:
```

```r
cat(sprintf("LM = %.3f\n", LM_stat))
```

```
## LM = 74.319
```

```r
cat(sprintf("p-valor = %.4f\n", p_valor_chi2))
```

```
## p-valor = 0.0000
```