Testes de Hipóteses em regressão linear

Renato

Testes de hipóteses

- Assunto amplo e com varias polemicas
- Existe uma teoria geral de testes de hipóteses. Veremos isso na segunda metade do curso.
- Como no caso de ICs, vamos no concentrar apenas nos principais testes de hipóteses associados com o modelo de regressão linear.
- Vamos começar com nosso exemplo básico da resistência à compressão de blocos de cimento.

Kaggle Dataset

- Aim: To predict the compressive strength of concrete based on material composition.

🎯 **Target Variable (Response Variable)**

Feature Name	Description	Units	Typical Range
Compressive Strength	The maximum compressive stress the concrete can withstand.	MPa (MegaPascals)	2.33 - 82.6

- Number of Samples: 1,030 observations
- Number of Features: 8 predictors

```
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

ICs de 95% para cada coeficiente

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

A variável “cement” tem um IC de 95% igual a

(0.103, 0.136)

Quando cement passa para cement+1, a resistência aumenta entre 0.103 e 0.136

A incerteza sobre o valor do coeficiente é refletida na largura do IC

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

A variável “water” tem um efeito negativo

IC = (-0.229, -0.071)

Quando water passa para water+1, a resistência DIMINUI entre 0.229 e 0.071

Predição com incerteza

No primeiro caso:

- embora com incerteza, predizemos um efeito positivo de “cement”

No segundo caso:

- com incerteza, predizemos um efeito negativo de “water”

Um terceiro caso: o IC contém o valor zero

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
------------------------------------------------------------------------------
```

IC de “fineaggregate”

IC = (-0.001, 0.041)

Quando fineaggregate passa para fineaggregate+1, a resistência pode diminuir ou aumentar.

A incerteza sobre o valor inclui incerteza até sobre a direção do efeito (positivo ou negativo)

ICs contendo zero

Os ICs contendo o valor zero formam uma classe especial:

- são coeficientes em que não temos confiança sobre seu efeito
- Pode ser positivo, negativo, pode ser ZERO.

O que acontece se o verdadeiro valor do coeficiente for ZERO?

Nesse caso, a variável pode ser descartada do modelo.

$$
\begin{aligned}
(Y|\mathbf{x}) &\sim N(\beta_0 + \beta_1 x_1 + \mathbf{0} x_2 + \beta_3 x_3, \sigma^2) \\
&\sim N(\beta_0 + \beta_1 x_1 + \beta_3 x_3, \sigma^2)
\end{aligned}
$$

Testando se verdadeiro coeficiente e’ zero

Uma maneira simples de descartar variáveis irrelevantes no modelo de regressão linear e’ olhar os ICs.

Se contém ZERO, a variável e’ descartada.

Esse método e’ muito simples e possui algumas desvantagens que veremos mais tarde.

Esse método e’ baseado num teste de hipótese de que o verdadeiro coeficiente e’ zero.

Além dos ICs, olhamos também os p-valores dos testes.

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
------------------------------------------------------------------------------
```

Os principais testes de hipóteses associados com regressão linear estão aqui.

Vamos agora conectar ICs a teoria de testes de hipóteses.

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 csMPa   R-squared:                       0.616
Model:                           OLS   Adj. R-squared:                  0.613
Method:                Least Squares   F-statistic:                     204.3
Date:               Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                       16:43:15   Log-Likelihood:                -3869.0
No. Observations:               1030   AIC:                             7756.
Df Residuals:                   1021   BIC:                             7800.
Df Model:                          8
Covariance Type:           nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

Assuma que o modelo de regressão linear e' o mecanismo gerador dos dados.

Existe um vetor desconhecido de coeficientes que gera os dados e que queremos aprender

$$\boldsymbol{\beta}^* = \begin{bmatrix} \beta_0^* \\ \beta_1^* \\ \vdots \\ \beta_p^* \end{bmatrix}$$

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
======================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
```

Sabemos que

$$\hat{\boldsymbol{\beta}} \sim N_{p+1}(\boldsymbol{\beta}^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1})$$

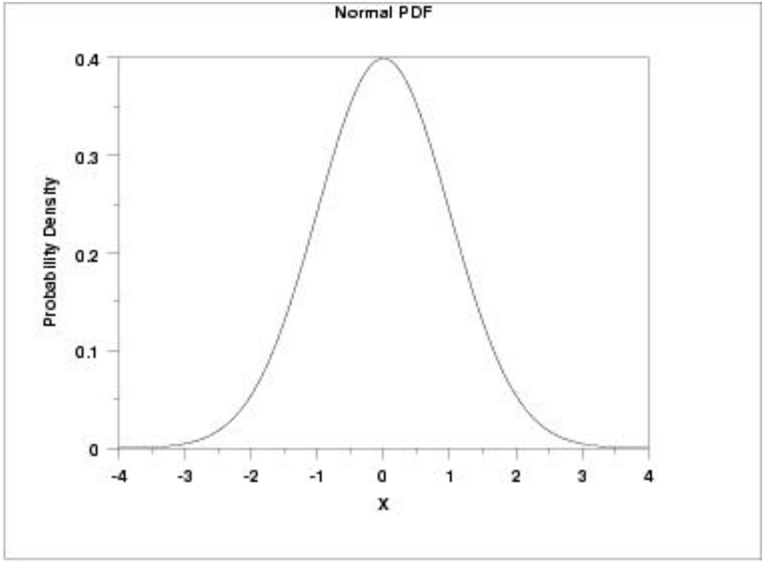
Então: cada coordenada $\hat{\beta}_j$ do vetor estimado $\hat{\boldsymbol{\beta}}$ oscila como uma Gaussiana em torno do seu verdadeiro e desconhecido valor β_j

$$\hat{\beta}_j \sim N(\beta_j^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj])$$

Gaussiana padrão

Mas sabemos muito sobre o comportamento de variáveis aleatórias Gaussianas.

1) Considere a Gaussiana padrão: N(0,1):
   a) Dificilmente sai de (-2, 2)
   b) Probab(estar em (-2,2)) = 0.95 (ou 95%)



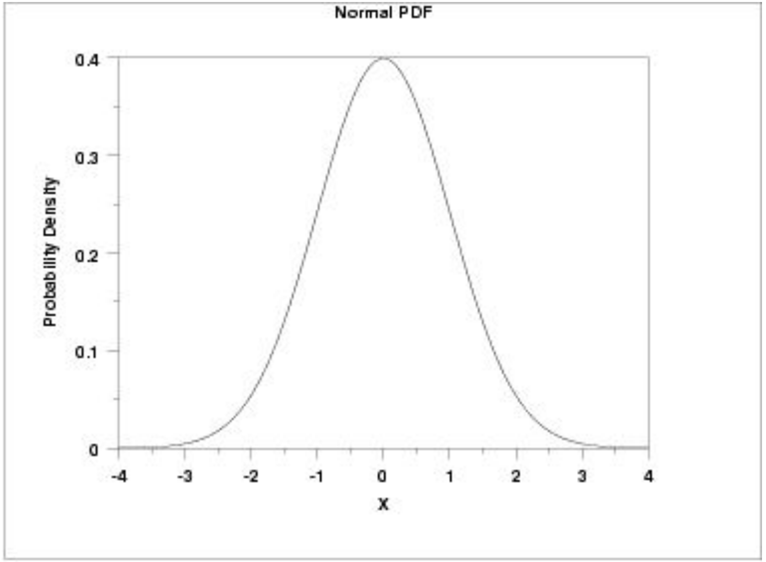
Gaussiana padrão

Mas sabemos muito sobre o comportamento de variáveis aleatórias Gaussianas.

1) Considere a Gaussiana padrão: N(0,1):
   a) Dificilmente sai de (-2, 2)
   b) Probab(estar em (-2,2)) = 0.95 (ou 95%)

2) Toda Gaussiana $X \sim N(\mu, \sigma^2)$

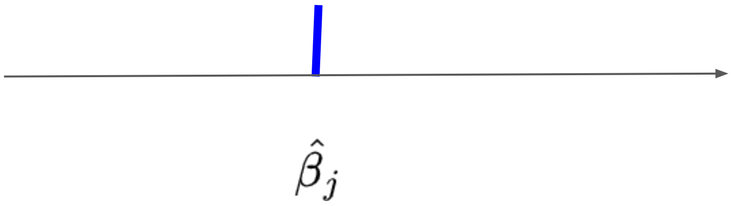
   pode ser transformada em Gaussiana padrão: $Z = \frac{X-\mu}{\sigma} \sim N(0,1)$



De volta para regressão linear…

$$\hat{\beta}_j \sim N(\beta_j^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj])$$

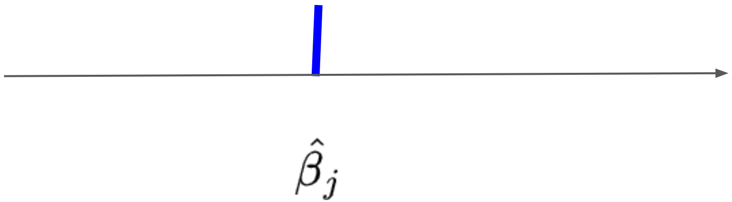
$$Z = \frac{\hat{\beta}_j - \beta_j^*}{\sqrt{\sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj]}} \sim N(0,1)$$



De volta para regressão linear…

$$\hat{\beta}_j \sim N(\beta_j^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj])$$

$$Z = \frac{\hat{\beta}_j - \beta_j^*}{\sqrt{\sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj]}} \sim N(0,1)$$



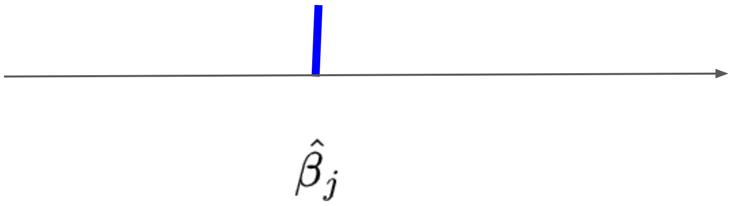
Esta distribuição e' a correta sempre que usarmos o VERDADEIRO e DESCONHECIDO valor de β_j^*

Com os ICs: como Z está entre -2 e 2 com alta probabilidade, revertemos a desigualdade para obter um intervalo de valores razoáveis para o desconhecido e verdadeiro β_j^*

De volta para regressão linear…

$$\hat{\beta}_j \sim N(\beta_j^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj])$$

$$Z = \frac{\hat{\beta}_j - \beta_j^*}{\sqrt{\sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj]}} \sim N(0,1)$$



FUNDAMENTAL: se subtrairmos um valor errado (ao invés do verdadeiro β_j^*) a distribuição de Z não vai seguir uma N(0,1)

Exemplo

Suponha que $\hat{\beta}_j \sim N(\beta_j^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj]) = N(5.5, 4.0)$

Então

$$Z = \frac{\hat{\beta}_j - 5.5}{\sqrt{4.0}} \sim N(0, 1)$$

O que acontece se usarmos um valor diferente do verdadeiro valor 5.5?

Por exemplo, se usarmos ZERO?

Exemplo

Suponha que $\hat{\beta}_j \sim N(\beta_j^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj]) = N(5.5, 4.0)$

Temos

$$Z = \frac{\hat{\beta}_j - 5.5}{\sqrt{4.0}} \sim N(0, 1)$$

O que acontece se usarmos um valor diferente do verdadeiro valor 5.5?

Por exemplo, se usarmos ZERO?

$$\frac{\hat{\beta}_j - 0}{\sqrt{4.0}} = \frac{\hat{\beta}_j \pm 5.5 - 0}{\sqrt{4.0}} = \frac{\hat{\beta}_j - 5.5}{\sqrt{4.0}} + \frac{(5.5 - 0)}{\sqrt{4.0}} = N(0, 1) + \frac{\beta_j^* - 0}{\sqrt{4.0}} = N(\frac{\beta_j^*}{2}, 1)$$

Resumo:

Suponha que $\hat{\beta}_j \sim N(\beta_j^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj]) = N(5.5, 4.0)$

Quando usamos o verdadeiro β_j^* no numerador, temos $Z = \frac{\hat{\beta}_j - 5.5}{\sqrt{4.0}} \sim N(0,1)$

Se usarmos o valor ZERO, teremos

$$\frac{\hat{\beta}_j - 0}{\sqrt{4.0}} \sim N(\frac{\beta_j^*}{\sqrt{4.0}}, 1)$$

Teremos esta quantidade como $\frac{\hat{\beta}_j - 0}{\sqrt{v^2}} \sim N(0,1)$ se, e somente se, $\beta_j^* = 0$

onde $v^2 = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj]$

Resumo 2:

Suponha que $\hat{\beta}_j \sim N(\beta_j^*, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}[jj]) = N(\beta_j^*, v^2)$

Se usarmos o valor ZERO, teremos $\frac{\hat{\beta}_j - 0}{\sqrt{v^2}} \sim N(\frac{\beta_j^*}{v}, 1)$

Se a HIPÓTESE H: $\beta_j^* = 0$ for verdadeira → temos N(0,1) → entre -2 e 2

Se a hipótese for falsa, a Gaussiana estará centrada num valor diferente de zero

Como usamos esse resultado na prática?

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
------------------------------------------------------------------------------------
```

$$t = \frac{\hat{\beta}_j - 0}{\sqrt{v^2}}$$

Se o verdadeiro coeficiente e' ZERO (variável pode ser descartada) então t deve estar entre -2 e 2

Se t estiver fora desse intervalo, e' evidência de que o verdadeiro e desconhecido β_j^* e' diferente de zero e variável NÃO DEVERIA ser descartada.

P-valor

```
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

E’ uma probabilidade

Supõe que a hipótese nula

$H_0 : \beta_j^* = 0$ e’ verdadeira

Nesse caso devemos ter

$$t = \frac{\hat{\beta}_j - 0}{\sqrt{v^2}} \sim N(0,1)$$

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 csMPa   R-squared:                       0.616
Model:                           OLS   Adj. R-squared:                  0.613
Method:                Least Squares   F-statistic:                     204.3
Date:               Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                       16:43:15   Log-Likelihood:                -3869.0
No. Observations:               1030   AIC:                             7756.
Df Residuals:                   1021   BIC:                             7800.
Df Model:                          8
Covariance Type:           nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

P-valor

Se $H_0 : \beta_j^* = 0$ e' verdadeira,

temos $t = \frac{\hat{\beta}_j - 0}{\sqrt{v^2}} \sim N(0,1)$

Observamos t=3.128 para superplasticizer, fora de (-2, 2)

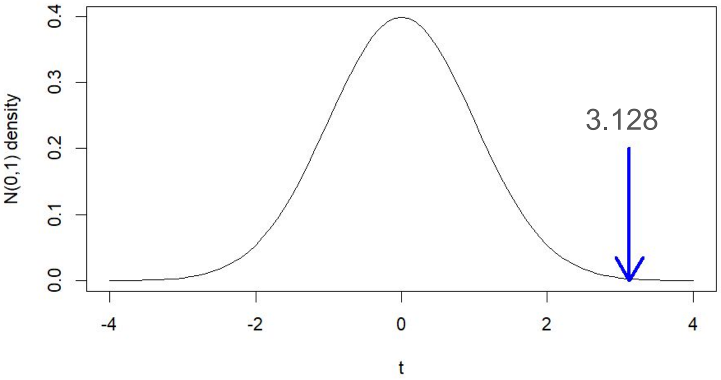
P-valor calcula quão extremo e' este valor observado CASO A HIPÓTESE NULA SEJA VERDADEIRA

P-valor

```
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 csMPa   R-squared:                       0.616
Model:                           OLS   Adj. R-squared:                  0.613
Method:                Least Squares   F-statistic:                     204.3
Date:               Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                       16:43:15   Log-Likelihood:                -3869.0
No. Observations:               1030   AIC:                             7756.
Df Residuals:                   1021   BIC:                             7800.
Df Model:                          8
Covariance Type:           nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```



P-valor

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

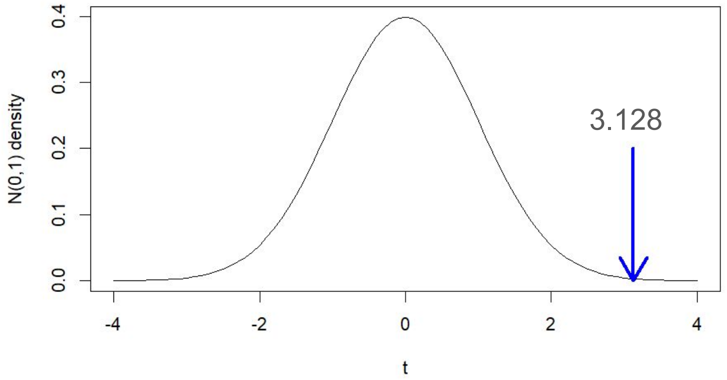
$$\text{p-valor} \;= \mathbb{P}(|t| > |t_{\text{obs}}|\,|H_0 \text{ e' verdadeira })$$

$$\mathbb{P}(|t| > 3.128|H_0 \text{ e' verdadeira }) = 0.002$$

OLS Regression Results

Dep. Variable:	csMPa	R-squared:	0.616
Model:	OLS	Adj. R-squared:	0.613
Method:	Least Squares	F-statistic:	204.3
Date:	Fri, 15 Oct 2021	Prob (F-statistic):	6.29e-206
Time:	16:43:15	Log-Likelihood:	-3869.0
No. Observations:	1030	AIC:	7756.
Df Residuals:	1021	BIC:	7800.
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-23.3312	26.586	-0.878	0.380	-75.500	28.837
cement	0.1198	0.008	14.113	0.000	0.103	0.136
slag	0.1039	0.010	10.247	0.000	0.084	0.124
flyash	0.0879	0.013	6.988	0.000	0.063	0.113
water	-0.1499	0.040	-3.731	0.000	-0.229	-0.071
superplasticizer	0.2922	0.093	3.128	0.002	0.109	0.476
coarseaggregate	0.0181	0.009	1.926	0.054	-0.000	0.037
fineaggregate	0.0202	0.011	1.887	0.059	-0.001	0.041
age	0.1142	0.005	21.046	0.000	0.104	0.125



Usando p-valor e testes

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

p-valor muito pequeno (menor que 0.05) → $\beta_j^* = 0$ nao e' compativel com os dados

→ REJEITAMOS a hipótese nula

$$H_0 : \beta_j^* = 0$$

P-valor grande (> 0.05): Hipótese nula $\beta_j^* = 0$ e' compatível com os dados.

→ ACEITAMOS a hipótese nula

Um detalhe

O valor do denominador da razão t precisa ser aprendido (σ^2 e’ desconhecido)

O cálculo exato do p-valor usa a distribuição t-Student com n-(p+1) graus de liberdade.

Se n-(p+1) > 30 e’ praticamente idêntico a usar a N(0,1)

Equivalências

Existem equivalências entre o resultado do teste de hipótese e IC:

- p-valor < 0.05 se, e somente se, 0 não pertence ao IC de 95%
- Se 0 não pertence ao IC de 95%, rejeita a hipótese nula $H_0 : \beta_j^* = 0$

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):         6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

Mais um teste

Testando se TODAS as features podem ser zeradas.

$$H_0 : \boldsymbol{\beta}^* = \mathbf{0}$$

$$H_0 : \begin{bmatrix} \beta_0^* \\ \beta_1^* \\ \vdots \\ \beta_p^* \end{bmatrix} = \begin{bmatrix} \beta_0^* \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement               0.1198      0.008     14.113      0.000       0.103       0.136
slag                 0.1039      0.010     10.247      0.000       0.084       0.124
flyash               0.0879      0.013      6.988      0.000       0.063       0.113
water               -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer     0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate      0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate        0.0202      0.011      1.887      0.059      -0.001       0.041
age                  0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

Como testar?

Se todas as features tem coeficientes zero, os resíduos com ou sem elas deveriam ser similares.

Regressão sem as features → fica apenas a coluna de 1's

Projeção de Y no espaço das combinações lineares do vetor (1, 1, …, 1):

$$\bar{y}(1, 1, \ldots, 1)^t$$

Two sum of residuals

```
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):         6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
==============================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const            -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement             0.1198      0.008     14.113      0.000       0.103       0.136
slag               0.1039      0.010     10.247      0.000       0.084       0.124
flyash             0.0879      0.013      6.988      0.000       0.063       0.113
water             -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer   0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate    0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate      0.0202      0.011      1.887      0.059      -0.001       0.041
age                0.1142      0.005     21.046      0.000       0.104       0.125
```

If the null hypothesis is true:

$$H_0 : \boldsymbol{\beta}^* = \mathbf{0}$$

then

$$y_i - \hat{y}_i \approx y_i - \bar{y}$$

Why? Because

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \ldots + \hat{\beta}_p x_p \approx \hat{\beta}_0 + 0x_1 + \ldots + 0x_p = \hat{\beta}_0$$

Compare the two sum of residuals:

$$\frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

Comparing lengths of vectors

```python
#generate OLS regression results for all features
import statsmodels.api as sm

X_sm = sm.add_constant(X)
model = sm.OLS(y,X_sm)
print(model.fit().summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  csMPa   R-squared:                       0.616
Model:                            OLS   Adj. R-squared:                  0.613
Method:                 Least Squares   F-statistic:                     204.3
Date:                Fri, 15 Oct 2021   Prob (F-statistic):          6.29e-206
Time:                        16:43:15   Log-Likelihood:                -3869.0
No. Observations:                1030   AIC:                             7756.
Df Residuals:                    1021   BIC:                             7800.
Df Model:                           8
Covariance Type:            nonrobust
=====================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------
const               -23.3312     26.586     -0.878      0.380     -75.500      28.837
cement                0.1198      0.008     14.113      0.000       0.103       0.136
slag                  0.1039      0.010     10.247      0.000       0.084       0.124
flyash                0.0879      0.013      6.988      0.000       0.063       0.113
water                -0.1499      0.040     -3.731      0.000      -0.229      -0.071
superplasticizer      0.2922      0.093      3.128      0.002       0.109       0.476
coarseaggregate       0.0181      0.009      1.926      0.054      -0.000       0.037
fineaggregate         0.0202      0.011      1.887      0.059      -0.001       0.041
age                   0.1142      0.005     21.046      0.000       0.104       0.125
==============================================================================
```

Two sum of residuals

Consider the degrees of freedom:

$$F = \frac{\sum_i (y_i - \hat{y}_i)^2 / (n - (p+1))}{\sum_i (y_i - \bar{y})^2 / (n-1)}$$

Ratio of two INDEPENDENT chi-squared distributions (divided by their degrees of freedom) has a KNOWN distribution: the F-distribution.

F for (Ronald) Fisher

https://en.wikipedia.org/wiki/F-distribution

https://www.nature.com/articles/s41437-020-00394-6#Bib1

Um voo mais filosófico: Karl Popper

1902 - 1994

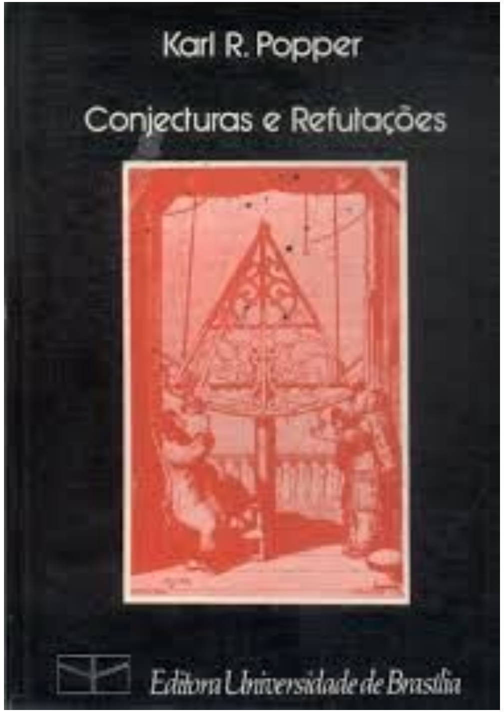
Viena → Nova Zelândia → Inglaterra (de 46 pra frente)

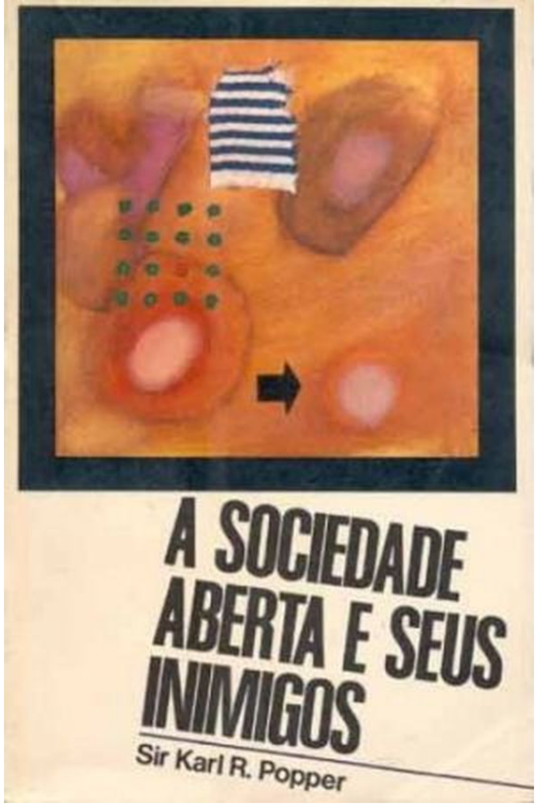
Marxista na juventude

O que torna uma teoria científica?

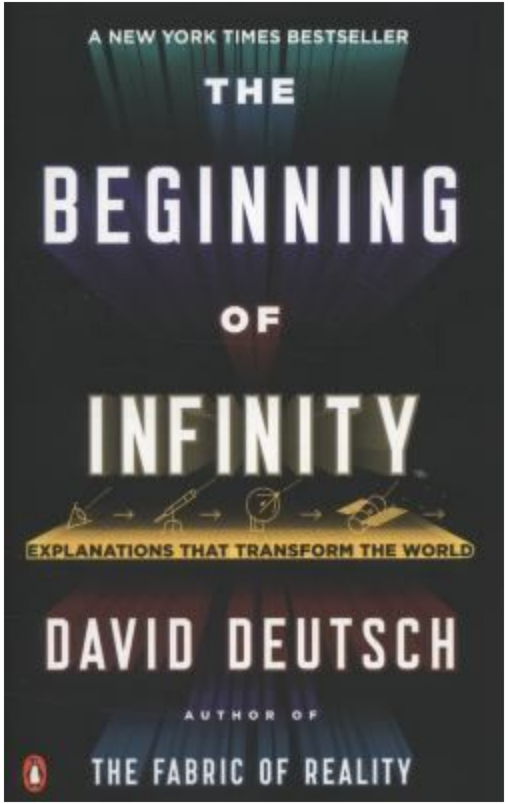
Psicanálise e marxismo são ciencia? A teoria de Einstein e' científica?

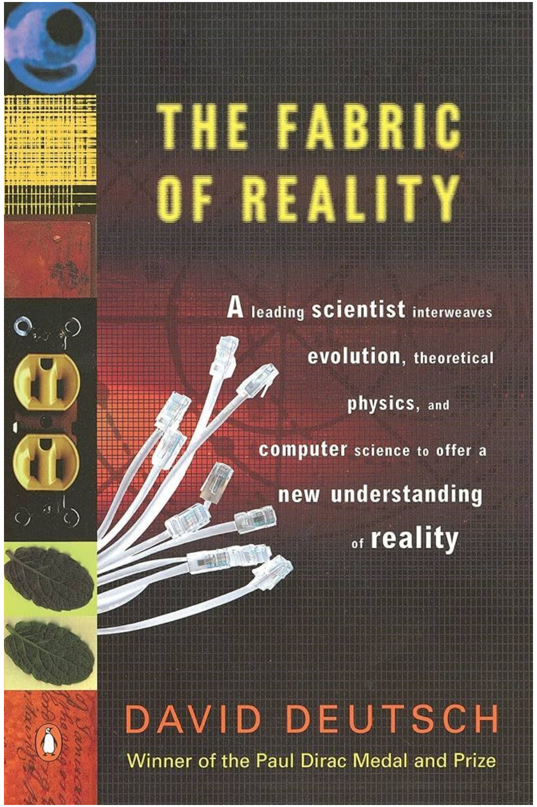
Livros em portugues

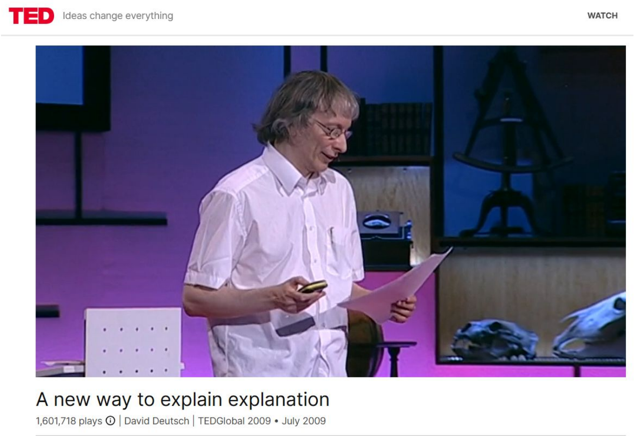




Porta-voz mais moderno: David Deutsch







https://www.ted.com/talks/david_deutsch_a_new_way_to_explain_explanation

Contexto

Teorias aparentemente perfeitas mostram-se erradas:

- Teoria Newtoniana durou séculos e é’ usada até hoje
- Mas foi um choque descobrir em 1920 (Einstein) que ela não era a explicação perfeita para o funcionamento do mundo físico.

- Psicanálise era um sucesso em Viena em 1920-1930. Era ciência?

- Como saber se uma teoria é científica?
- Como saber que uma teoria é correta?

Princípio da Falseabilidade

Uma forma de demarcar a ciência da não-ciência.

Uma teoria é científica se ela pode ser provada falsa.

Teste

1919: Sir Arthur Stanley Eddington and Frank Watson Dyson led two expeditions to observe a total solar eclipse. (Africa and SOBRAL, CE)

To measure how much starlight bends as it passes close to the sun.

The results confirmed Albert Einstein's theory of general relativity (light is curved by gravity)

It made Einstein a worldwide celebrity.

The main point: if the results were not confirmed, the theory was completely wrong, it would be FALSE.


HR1403
Taurus
HR1375
67 Tauri
65 Tauri
72 Tauri
69 Tauri

Princípio da Falseabilidade

Uma forma de demarcar a ciência da não-ciência.

Uma teoria é científica se ela pode ser provada falsa.

Para ser científica, a teoria deve ser:

- passível de ser testada
- passível de ser refutada
- se for refutada, a teoria está errada, e’ falsa
- se não for refutada, não quer dizer que seja verdadeira.

Conjecturas e refutações

Nosso conhecimento científico não é aquilo que sabemos ser verdade

Mas sim, o conjunto de teorias que não conseguimos refutar.

Teorias que não podem ser testadas dessa forma não são necessariamente absurdas, mas não são científicas.

E de onde vêm as ideias científicas? Não existe um método científico para gerar ideias e teorias.