

```
In [1]: import seaborn as sns
import statsmodels.formula.api as smf
import numpy as np
import pandas as pd
```

```
In [2]: titanic = sns.load_dataset("titanic")
titanic.head(3)
```

```
Out[2]:
```

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who
0	0	3	male	22.0	1	0	7.2500	S	Third	man
1	1	1	female	38.0	1	0	71.2833	C	First	woman
2	1	3	female	26.0	0	0	7.9250	S	Third	woman

Interpretação das chances de uma regressão logística

Interpretação do Intercepto

```
In [3]: model = smf.logit(formula='survived ~ 1', data=titanic).fit();
print(model.params)
```

```
Optimization terminated successfully.
      Current function value: 0.665912
      Iterations 4
Intercept    -0.473288
dtype: float64
```

$$\log\left(\frac{p}{1-p}\right) = -0.473288$$

```
In [4]: log_odd = model.params['Intercept']
odds = np.exp(log_odd)
p = odds/(1+odds)

f'Probabilidade p = {p:.2%}'
```

```
Out[4]: 'Probabilidade p = 38.38%'
```

Como queremos explicar a sobrevivência pelo intercepto...

```
In [5]: f'A chance de sobreviver é {p:.2%}'
```

```
Out[5]: 'A chance de sobreviver é 38.38%'
```

Outra forma de verificar

```
In [6]: sobrevivencia = pd.DataFrame({
        'Sim/Não': titanic['survived'].value_counts(),
        '%': titanic['survived'].value_counts(normalize=True)
    })

sobrevivencia['%'] = sobrevivencia['%'].map(lambda x: f'{x:.2%}')
sobrevivencia
```

```
Out [6]:
```

	Sim/Não	%
survived		
0	549	61.62%
1	342	38.38%

Interpretação de Variável Contínua

```
In [7]: model = smf.logit(formula='survived ~ fare', data=titanic).fit()
model.params
```

Optimization terminated successfully.
Current function value: 0.627143
Iterations 6

```
Out [7]: Intercept    -0.941330
fare                0.015197
dtype: float64
```

```
In [8]: log_odd = model.params['Intercept']
odds = np.exp(log_odd)
```

f'Para um passageiro que não pagou nada, as chances de sobreviver são de

```
Out [8]: 'Para um passageiro que não pagou nada, as chances de sobreviver são de
0.39%'
```

```
In [9]: log_odd = model.params['fare']
odds = np.exp(log_odd)
```

f'Para cada dólar adicional gasto, as chances de sobreviver aumentam em {

```
Out [9]: 'Para cada dólar adicional gasto, as chances de sobreviver aumentam em
1.02%'
```

Interpretação de Variável Categórica

```
In [10]: model = smf.logit(formula='survived ~ C(pclass)', data=titanic).fit()
model.params
```

Optimization terminated successfully.
Current function value: 0.607805
Iterations 5

```
Out [10]: Intercept                0.530628
C(pclass) [T.2]                  -0.639431
C(pclass) [T.3]                  -1.670399
dtype: float64
```

```
In [11]: log_odd = model.params['Intercept']  
         odds_i = np.exp(log_odd)  
  
         f'Para um passageiro de primera classe, as chances de sobreviver são de {'
```

```
Out[11]: 'Para um passageiro de primera classe, as chances de sobreviver são de  
         1.70%'
```

```
In [12]: log_odd = model.params['C(pclass)[T.2]']  
         odds_2nd = np.exp(log_odd)  
  
         f'Para um passageiro de segunda classe, as chances de sobreviver são as c
```

```
Out[12]: 'Para um passageiro de segunda classe, as chances de sobreviver são as c  
         hances de sobreviver na primeira classe multiplicado por 0.5275924802910  
         862'
```

```
In [13]: f'Para um passageiro de segunda classe, as chances de sobreviver são de {'
```

```
Out[13]: 'Para um passageiro de segunda classe, as chances de sobreviver são de  
         0.90%'
```