

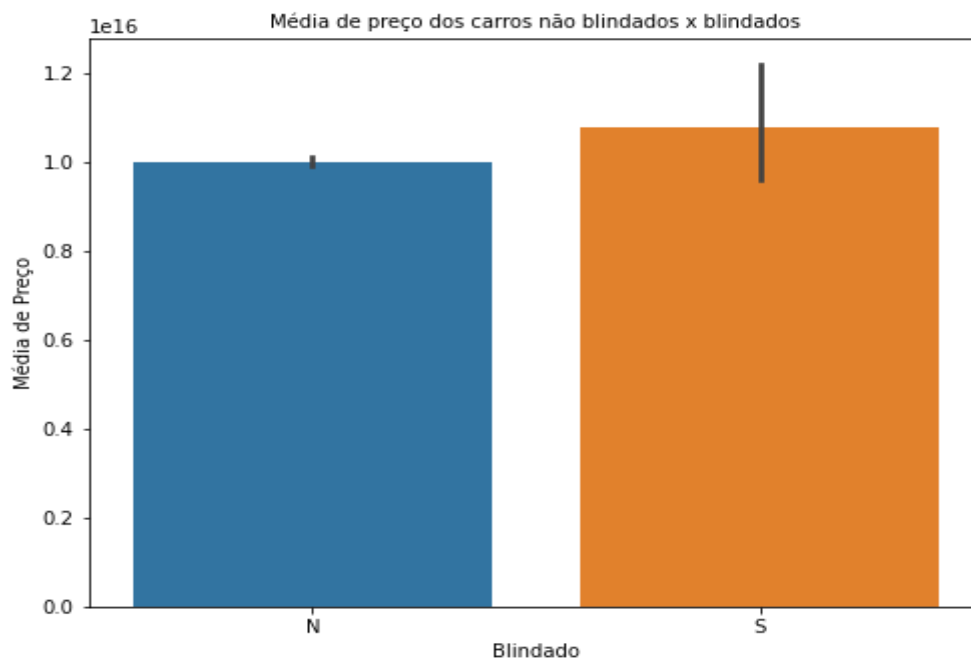
Processo para gerar previsão de Preço

```
train_ =  
train.drop(['modelo', 'versao', 'cidade_vendedor'], axis=1)
```

```
train.info()
```

```
train_ = train_.drop('veiculo_alienado', axis=1)  
train_.blindado.value_counts()
```

```
# Plot do gráfico  
plt.figure(figsize=(8, 6))  
sns.barplot(data=train_, x='blindado', y='preco')  
# Configurações do gráfico  
plt.xlabel('Blindado', fontsize=10)  
plt.ylabel('Média de Preço', fontsize=10)  
plt.title('Média de preço dos carros não blindados x blindados',  
          fontsize=10)  
# Exibir o gráfico  
plt.show()
```



```
train.blindado = train.blindado.replace('N', 0)
```

```
train.blindado = train.blindado.replace('S',1)
train.blindado.value_counts()
```

```
train.describe()
```

	num_fotos	ano_de_fabricacao	ano_modelo	odometro	num_portas	blindado	ga
count	29407.000000	29584.000000	29584.000000	29584.000000	29584.000000	29584.000000	
mean	10.323834	2016.758552	2017.808985	58430.592077	3.940677	0.008383	
std	3.487334	4.062422	2.673930	32561.769309	0.338360	0.091175	
min	8.000000	1985.000000	1997.000000	100.000000	2.000000	0.000000	
25%	8.000000	2015.000000	2016.000000	31214.000000	4.000000	0.000000	
50%	8.000000	2018.000000	2018.000000	57434.000000	4.000000	0.000000	
75%	14.000000	2019.000000	2020.000000	81953.500000	4.000000	0.000000	
max	21.000000	2022.000000	2023.000000	390065.000000	4.000000	1.000000	

```
train.describe().transpose()
```

	count	mean	std	min	25%	50%	75%	max
num_fotos	29407.0	1.032383e+01	3.487334e+00	8.000000e+00	8.000000e+00	8.000000e+00	1.400000e+01	2.100000e+01
ano_de_fabricacao	29584.0	2.016759e+03	4.062422e+00	1.985000e+03	2.015000e+03	2.018000e+03	2.019000e+03	2.022000e+03
ano_modelo	29584.0	2.017809e+03	2.673930e+00	1.997000e+03	2.016000e+03	2.018000e+03	2.020000e+03	2.023000e+03
odometro	29584.0	5.843059e+04	3.256177e+04	1.000000e+02	3.121400e+04	5.743400e+04	8.195350e+04	3.900650e+05
num_portas	29584.0	3.940677e+00	3.383603e-01	2.000000e+00	4.000000e+00	4.000000e+00	4.000000e+00	4.000000e+00
blindado	29584.0	8.382910e-03	9.117520e-02	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00
garantia_de_fabrica	29584.0	1.475460e-01	3.546553e-01	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00
revisoes_dentro_agenda	29584.0	1.997701e-01	3.998343e-01	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00
veiculo_alienado	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
preco	29584.0	1.002299e+16	9.547504e+15	2.006292e+10	3.196583e+15	8.244891e+15	1.342533e+16	6.549912e+16

```
train_.veiculo_único_dono.value_counts()
```

```
Único dono      10423
Name: veiculo_único_dono, dtype: int64
```

```
train_.veiculo_único_dono =
train_.veiculo_único_dono.replace(train.veiculo_único_dono.value
_counts().index[0],1)
train_.veiculo_único_dono = train_.veiculo_único_dono.fillna(0)
```

```
train_.veiculo_único_dono.value_counts().sum()
```

```
train_.marca_popular = train_.marca_popular.astype(int)
```

```
train_.blindado.value_counts()
```

```
N    29336
S      248
Name: blindado, dtype: int64
```

```
train_.blindado = train_.blindado.replace('N',0)
train_.blindado = train_.blindado.replace('S',1)
train_.blindado.value_counts()
```

```
train_.revisoes_concessionaria.value_counts()
```

```
Todas as revisões feitas pela concessionária    9172
Name: revisoes_concessionaria, dtype: int64
```

```
train_.revisoes_concessionaria =
train_.revisoes_concessionaria.replace('Todas as revisões feitas
pela concessionária',1)
train_.revisoes_concessionaria =
train_.revisoes_concessionaria.fillna(0)
train_.revisoes_concessionaria.value_counts()
```

```
0.0    20412
1.0     9172
Name: revisoes_concessionaria, dtype: int64
```

```
train_.veiculo_licenciado.value_counts()
```

```
Licenciado    15906
Name: veiculo_licenciado, dtype: int64
```

```
train_.veiculo_licenciado =
train_.veiculo_licenciado.replace('Licenciado',1)
train_.veiculo_licenciado = train_.veiculo_licenciado.fillna(0)
train_.veiculo_licenciado.value_counts()
```

```
1.0    15906
0.0    13678
Name: veiculo_licenciado, dtype: int64
```

```
train_.num_fotos =
train_.num_fotos.fillna(train_.num_fotos.value_counts().index[0]
)
```

```
train_.entrega_delivery=train_.entrega_delivery.astype(int)
```

```
train_.troca = train_.troca.astype(int)
```

```
train_.elegivel_revisao = train_.elegivel_revisao.astype(int)
```

```
train.info()
```

```
regiao = smf.ols('preco ~
marca_popular+tipo+hodometro+veiculo_único_dono',train_).fit()
```

```
regiao.summary()
```

OLS Regression Results			
Dep. Variable:	preco	R-squared:	0.006
Model:	OLS	Adj. R-squared:	0.006
Method:	Least Squares	F-statistic:	20.07
Date:	Sun, 16 Jul 2023	Prob (F-statistic):	4.74e-34
Time:	15:45:38	Log-Likelihood:	-1.1304e+06
No. Observations:	29584	AIC:	2.261e+06
Df Residuals:	29574	BIC:	2.261e+06
Df Model:	9		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
Intercept	6.446e+15	1.87e+15	3.451	0.001	2.78e+15	1.01e+16
tipo[T.Hatchback]	4.799e+15	1.88e+15	2.558	0.011	1.12e+15	8.48e+15
tipo[T.Minivan]	9.972e+14	4.06e+15	0.246	0.806	-6.95e+15	8.95e+15
tipo[T.Perua/SW]	6.351e+15	2.62e+15	2.423	0.015	1.21e+15	1.15e+16
tipo[T.Picape]	5.394e+15	1.88e+15	2.876	0.004	1.72e+15	9.07e+15
tipo[T.Sedã]	4.273e+15	1.87e+15	2.284	0.022	6.06e+14	7.94e+15
tipo[T.Utilitário esportivo]	5.703e+15	1.88e+15	3.041	0.002	2.03e+15	9.38e+15
marca_popular	-7.6e+14	1.26e+14	-6.037	0.000	-1.01e+15	-5.13e+14
odometro	-9.222e+09	1.92e+09	-4.810	0.000	-1.3e+10	-5.46e+09
veiculo_único_dono	-2.756e+14	1.3e+14	-2.123	0.034	-5.3e+14	-2.11e+13
Omnibus:	15732.444	Durbin-Watson:	1.996			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	131756.828			
Skew:	2.445	Prob(JB):	0.00			
Kurtosis:	12.109	Cond. No.	6.39e+06			

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The condition number is large, 6.39e+06. This might indicate that there are strong multicollinearity or other numerical problems.

Para fazer a previsão do preço a partir dos dados apresentados, seria necessário entender o contexto e as características dos dados, bem como o objetivo da previsão. No entanto, posso fornecer algumas informações e análises com base nos resultados apresentados no resumo da regressão linear (OLS Regression Results).

Variáveis Utilizadas e Transformações:

As variáveis independentes (ou recursos) utilizadas para prever o preço incluem: 'tipo', 'marca_popular', 'odometro' e 'veiculo_único_dono'.

Algumas variáveis foram transformadas em variáveis categóricas, como 'tipo' e 'marca_popular', para representar diferentes categorias de veículos.

A variável 'veiculo_único_dono' parece ser uma variável binária, possivelmente indicando se o veículo possui apenas um dono anterior.

A variável 'odometro' parece representar o hodômetro do veículo, provavelmente em unidades desconhecidas.

Tipo de Problema:

Com base nos resultados apresentados, estamos resolvendo um problema de regressão, ou seja, estamos tentando prever o preço de um veículo com base em determinadas características.

Modelo Utilizado:

O modelo utilizado foi uma regressão linear (OLS - Ordinary Least Squares).

Os coeficientes associados a cada variável independente fornecem uma estimativa do impacto que cada variável tem no preço do veículo.

O valor p ($P > |t|$) associado a cada coeficiente indica a significância estatística da relação entre cada variável independente e o preço do veículo. Valores baixos de p indicam uma relação estatisticamente significativa.

Medida de Desempenho do Modelo:

O coeficiente de determinação (R-squared) indica a proporção da variabilidade do preço que é explicada pelo modelo. Nesse caso, o valor de R-squared é 0.006, o que indica que apenas 0.6% da variabilidade do preço é explicada pelas variáveis independentes incluídas no modelo.

Esse valor baixo de R-squared sugere que o modelo não está se ajustando bem aos dados ou que as variáveis incluídas podem não estar capturando adequadamente os padrões e relações no conjunto de dados.

Em termos de escolha do melhor modelo, seria necessário realizar uma análise mais aprofundada e considerar outras métricas de desempenho, como erro médio quadrático (RMSE), validação cruzada e comparação com outros modelos. Além disso, a seleção das variáveis e possíveis transformações ou engenharia de recursos também é uma etapa importante na construção de um modelo preditivo robusto.