# Seguro_medico_regressao

January 27, 2021

[2]:
```
import pandas as pd
data = pd.read_csv('insurance.csv')
```

Dados retirados de https://www.kaggle.com/mirichoi0218/insurance. O objetivo desse pequeno projeto é analisar os dados e indicar quais as variáveis que mais impactam no custo do seguro médico. Além disso, iremos utilizar algoritmos de regressão para prevermos futuros valores de custo de acordo com as características do sujeito em questão. Iremos discutir um pouco mais a aplicação de algoritmos de regressão quando chegarmos nesse ponto.

[31]:
```
data.shape
```

[31]:
```
(1338, 7)
```

[3]:
```
data.head() #visualizando os primeiros dados
```

[3]:

| | age | sex | bmi | children | smoker | region | charges |
|---|---|---|---|---|---|---|---|
| 0 | 19 | female | 27.900 | 0 | yes | southwest | 16884.92400 |
| 1 | 18 | male | 33.770 | 1 | no | southeast | 1725.55230 |
| 2 | 28 | male | 33.000 | 3 | no | southeast | 4449.46200 |
| 3 | 33 | male | 22.705 | 0 | no | northwest | 21984.47061 |
| 4 | 32 | male | 28.880 | 0 | no | northwest | 3866.85520 |

[4]:
```
data.columns
```

[4]:
```
Index(['age', 'sex', 'bmi', 'children', 'smoker', 'region', 'charges'],
dtype='object')
```

[5]:
```
data.columns = ['Idade', 'Sexo', 'Imc', 'Filhos', 'Fumante', 'Região', 'Custo']␣
↪#traduzindo as colunas
```

[6]:
```
data.head()
```

[6]:

| | Idade | Sexo | Imc | Filhos | Fumante | Região | Custo |
|---|---|---|---|---|---|---|---|
| 0 | 19 | female | 27.900 | 0 | yes | southwest | 16884.92400 |
| 1 | 18 | male | 33.770 | 1 | no | southeast | 1725.55230 |
| 2 | 28 | male | 33.000 | 3 | no | southeast | 4449.46200 |
| 3 | 33 | male | 22.705 | 0 | no | northwest | 21984.47061 |
| 4 | 32 | male | 28.880 | 0 | no | northwest | 3866.85520 |

[7]:
```
dic1 = {'southwest':'sudoeste','southeast':'sudeste','northwest':
 ↪'noroeste','northeast':'nordeste'}
data['Região'] = data['Região'].map(dic1) #traduzindo as regiões
```

[8]:
```
dic2 = {'male':'homem','female':'mulher'}
data['Sexo'] = data['Sexo'].map(dic2) #traduzindo o sexo
```

[9]:
```
dic3 = {'yes':'sim','no':'não'}
data['Fumante'] = data['Fumante'].map(dic3)
```

[10]:
```
data.head() #dados traduzidos e binarizados
```

[10]:

| | Idade | Sexo | Imc | Filhos | Fumante | Região | Custo |
|---|---|---|---|---|---|---|---|
| 0 | 19 | mulher | 27.900 | 0 | sim | sudoeste | 16884.92400 |
| 1 | 18 | homem | 33.770 | 1 | não | sudeste | 1725.55230 |
| 2 | 28 | homem | 33.000 | 3 | não | sudeste | 4449.46200 |
| 3 | 33 | homem | 22.705 | 0 | não | noroeste | 21984.47061 |
| 4 | 32 | homem | 28.880 | 0 | não | noroeste | 3866.85520 |

Para verificarmos as variáveis que mais impactam no Custo, primeiro devemos pensar mais profundamente se os dados fornecidos aqui de fato nos permitem responder a pergunta feita. Em uma primeira avaliação, é possível perceber que podemos responder a essa pergunta de forma particular. Ou seja, podemos indicar quais das variáveis **aqui presentes** mais impactam no Custo. Se nos fosse questionado se outra variável (presença ou não de tatuagens no corpo, por exemplo) impacta significativamente o Custo, não estaríamos aptos a responder com precisão.

Vamos fazer agora algumas considerações iniciais e, previamente, tentar estipluar uma ordem de impacto no Custo para as variáveis, onde a 1ª variável impacta mais que a 2ª e assim sucessivamente.

1ª: Imc, já que é sabido que a obesidade impacta profundamente as vidas das pessoas (basta observarmos que não vemos frequentemente homens e mulheres obesos comemorando seu 70º aniversário, por exemplo);

2ª: Idade, pois a probabilidade de desenvolvermos doenças aumenta à medida que vivemos mais. Note que uma coisa não causa a outra, mas convenhamos que se alguém vive 20 anos apenas, essa pessoa provavelmente morreu em um acidente fatal ou nasceu com alguma doença grave. Pessoas que vivem mais têm mais chance de serem acometidas por doenças simplesmente porque a probabilidade aumenta à medida que vivemos mais. Um exemplo claro disso é o aumento massivo de casos de câncer, onde, se deixarmos de lado, apenas por um segundo, a toxicidade da comida que ingerimos, podemos perceber claramente que o motivo pelo qual as pessoas não desenvolviam câncer há 150 anos atrás era porque elas não viviam tempo suficiente para que a doença começasse a se desenvolver;

3ª: Fumante, pois pessoas que fumam estão muito mais propensas a terem problemas respiratórios e desenvolver câncer. Classificamos a Idade como um fator de mais impacto pois é possível vermos pessoas fumantes atingirem idades mais avançadas, seja porque começaram a fumar quando mais velhas ou porque o cigarro danifica os pulmões mais lentamente;

4ª: Número de filhos, pois quanto mais dependentes uma pessoa tem, maiores as chances dela precisar utilizar o seguro médico em algum momento;

5ª: Região, pois uma dada região pode conter mais pessoas com um estado ruim de saúde. Aqui é importante ressaltar que não estamos considerando que a região afeta a saúde (ou o oposto), primeiro porque, para fazermos tal suposição, precisaríamos de mais dados para constatar que uma dada região afeta a saúde das pessoas (por fatorem ambientais, como falta de saneamento básico, por exemplo). E segundo, pois é muito mais provável que uma região simplesmente **contenha** pessoas que fizeram péssimas escolhas quando se trata de sua saúde, fato esse que pode ser cultural, inclusive;

6ª: Sexo, pois homens e mulheres têm chances similares de serem acometidos por doenças (ignorando doenças seletivas, é claro). A mulher tem o caso particular da gravidez e do parto, porém aqui iremos considerar que o custo é igualmente dividido entre o casal.

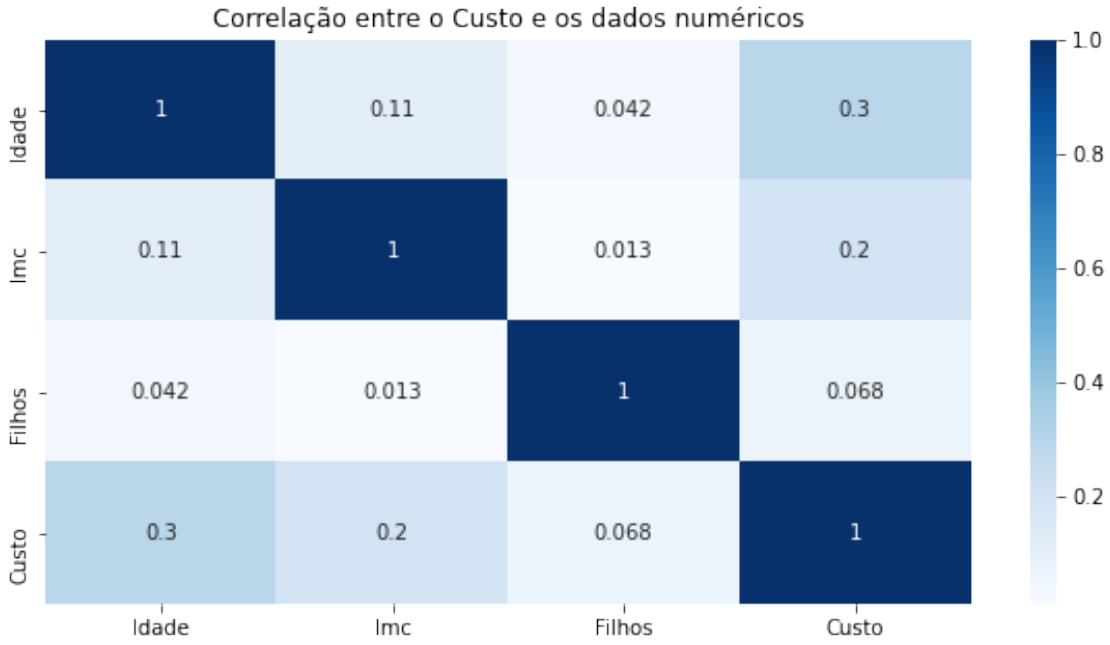
[11]:
```
#Verificando dados nulos em porcentagem:
data.isnull().sum()*100/float(len(data))
```

[11]:
```
Idade      0.0
Sexo       0.0
Imc        0.0
Filhos     0.0
Fumante    0.0
Região     0.0
Custo      0.0
dtype: float64
```

[12]:
```
#Verificando os tipos das variáveis:
data.dtypes
```

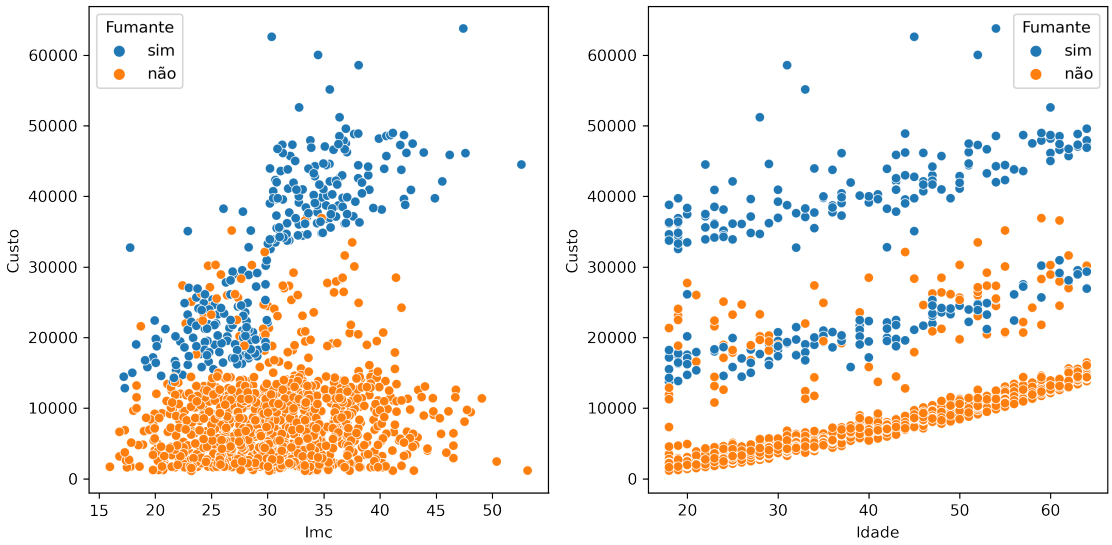
[12]:
```
Idade        int64
Sexo        object
Imc        float64
Filhos       int64
Fumante     object
Região      object
Custo      float64
dtype: object
```

[13]:
```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(10,5))
sns.heatmap(data.corr(),annot=True,cmap='Blues')
plt.title('Correlação entre o Custo e os dados numéricos')
plt.show()

#Vemos que a Idade impacta positivamente o Custo, assim como o IMC, porém com␣
 ↪uma taxa maior.
#Já o número de Filhos tem impacto muito baixo.
```



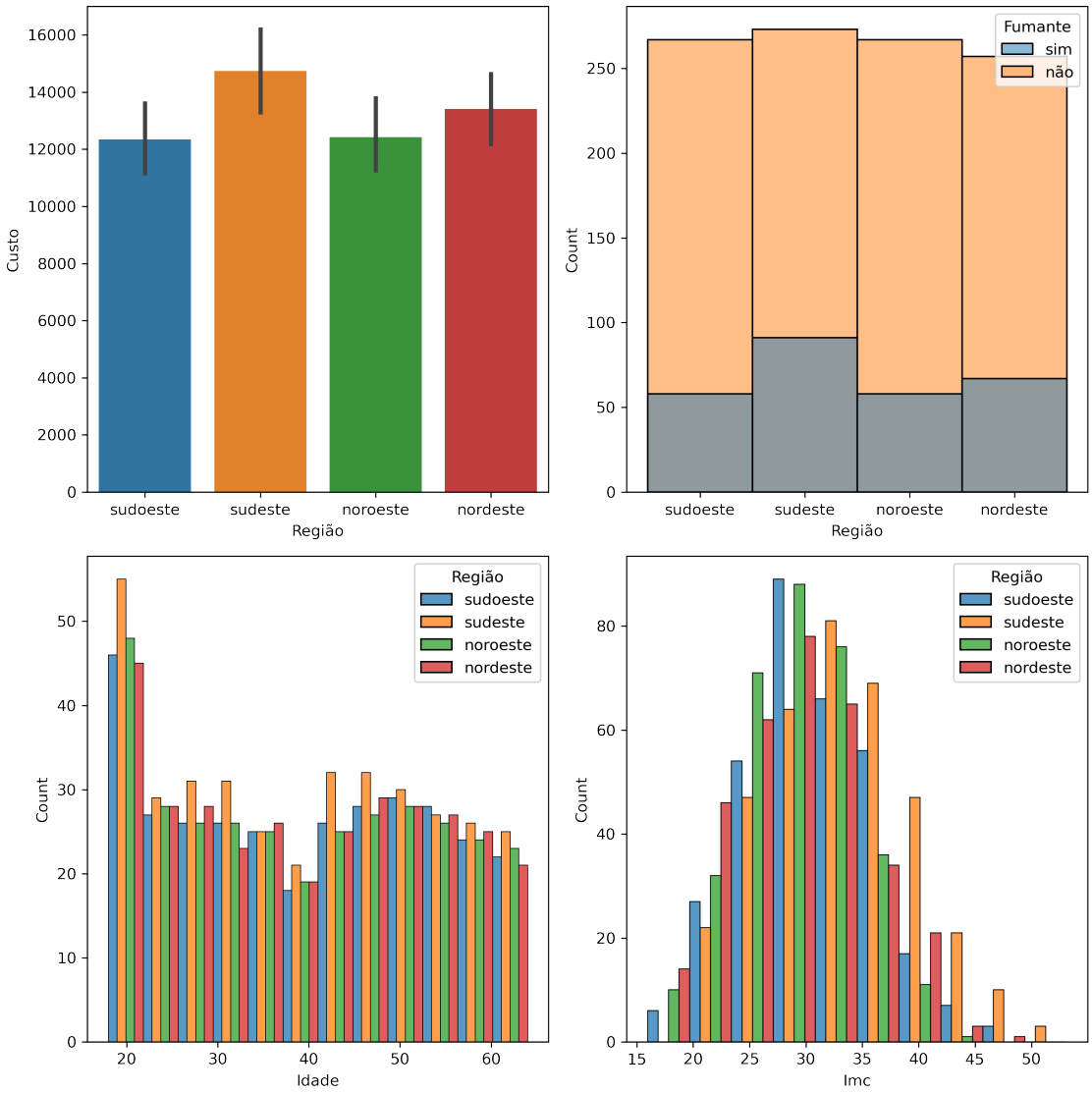

Como podemos perceber, o mapa de calor acima não nos mostra a correlação entre as variáveis qualitativas (Fumante, Sexo e Região), já que a correlação nos fornece informações somente sobre variáveis quantitativas. Desse modo, precisamos verificar graficamente se as variáveis qualitativas impactam no Custo. Primeiramente, vamos verificar o efeito do cigarro no Custo quando combinamos a droga ao Imc e à Idade.

[14]:
```
plt.figure(figsize=(10,5),dpi=300)
plt.subplot(1,2,1)
sns.scatterplot(x=data['Imc'],y=data['Custo'],hue=data['Fumante'])

plt.subplot(1,2,2)
sns.scatterplot(x=data['Idade'],y=data['Custo'],hue=data['Fumante'])

plt.tight_layout()
```




Ambos os gráficos nos confirmam que a Idade é um fator de alta relevância para o Custo e, quando combinado com o cigarro, pessoas de mesma idade pagam consideravelmente mais se fumam. Agora vamos colocar lado a lado as características ligadas a saúde (Idade, Imc e Fumante) por Região.

[16]:
```
plt.figure(figsize=(10,10),dpi=300)
plt.subplot(2,2,1)
sns.barplot(data=data,x=data['Região'],y=data['Custo'])

plt.subplot(2,2,2)
sns.histplot(data=data,x=data['Região'],hue=data['Fumante'])

plt.subplot(2,2,3)
sns.histplot(data=data,x=data['Idade'],hue=data['Região'],multiple='dodge')

plt.subplot(2,2,4)
sns.
 ↪histplot(data=data,x=data['Imc'],hue=data['Região'],multiple='dodge',bins=10)


plt.tight_layout()
plt.show()
```

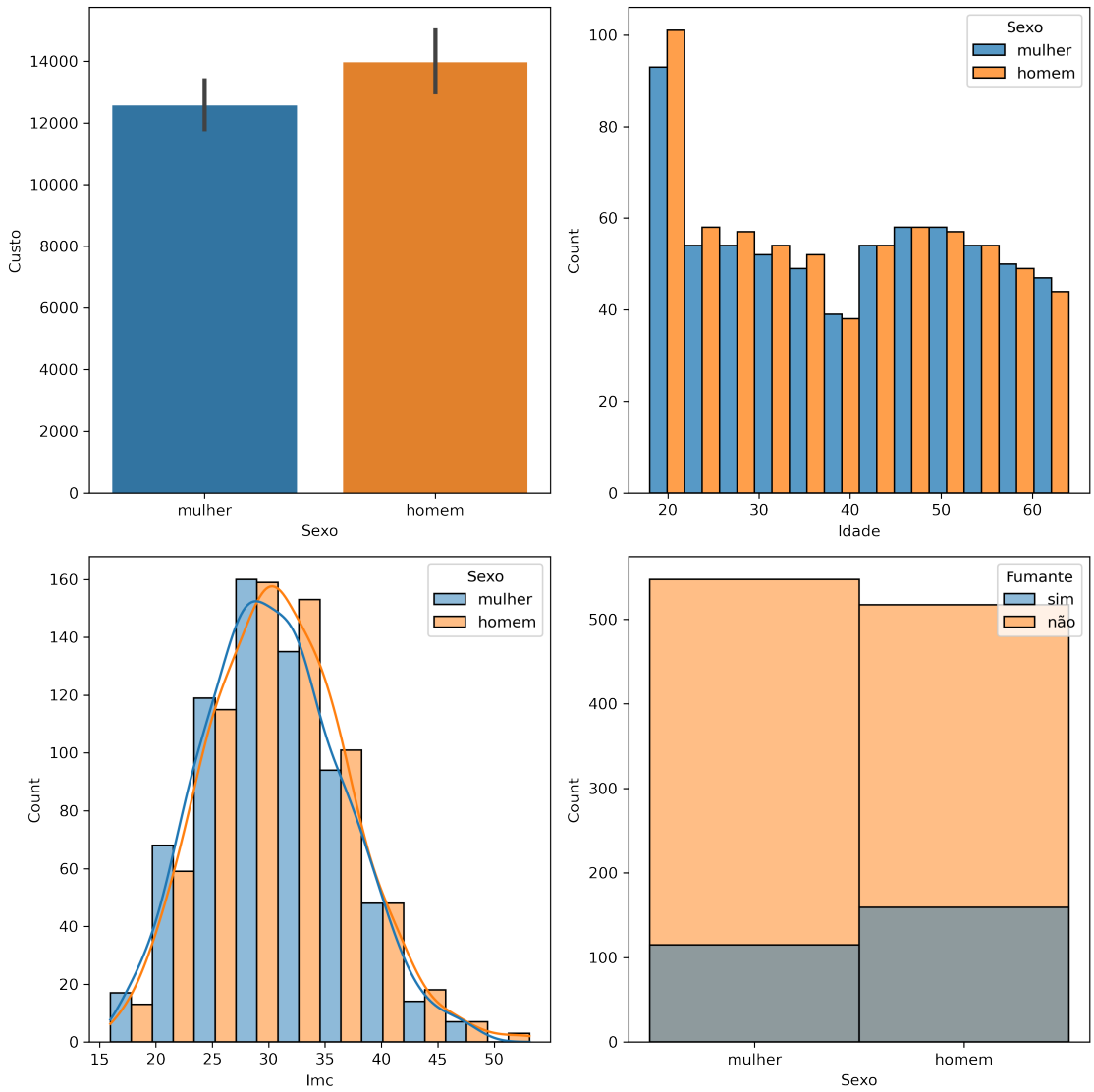



Podemos ver que a região Sudeste é a que tem o Custo mais alto e isso é gerado pelo maior número de fumantes, pela prevalência em quase todos os intervalos de Idade e pela predominância nos altos valores de Imc. Isso confirma, ao menos parcialmente, nossa hipótese de que uma região teria um Custo maior pois as pessoas nesta fizeram escolhas ruins no que se trata de sua saúde.

Quanto ao Sexo, podemos verificar se homens ou mulheres têm um Custo maior e, na sequência, comparar ambos os Sexos no que se trata da Idade, Imc e Fumante para verificarmos se hábitos ruins geram o Custo mais elevado.

[15]:
```
plt.figure(figsize=(10,10),dpi=300)
plt.subplot(2,2,1)
sns.barplot(data=data,x=data['Sexo'],y=data['Custo'])

plt.subplot(2,2,2)
```

```
sns.histplot(data=data,x=data['Idade'],hue=data['Sexo'],multiple='dodge')

plt.subplot(2,2,3)
sns.
 ↪histplot(data=data,x=data['Imc'],hue=data['Sexo'],multiple='dodge',bins=10,kde=True)

plt.subplot(2,2,4)
sns.histplot(data=data,x=data['Sexo'],hue=data['Fumante'])


plt.tight_layout()
plt.show()
```



Podemos perceber que homens têm um Custo mais elevado, o que pode ser explicado pelo maior número de fumantes e pelo Imc levemente mais elevado. No que se trata da Idade, há um equilíbrio bastante grande, então o Custo não é muito afetado de forma particular para cada Sexo. Para confirmarmos nosso raciocínio feito a partir dos gráficos, vamos calcular os coeficientes de variação para a Idade e Imc, de acordo com o Sexo.

O coeficiente de variação de uma dada variável é dado por $C_V = \frac{\sigma}{\mu}$ e nos indica o quão concentrados em torno da média os dados estão. Quanto menor o coeficiente de variação, mais homogênea é a amostra.

[16]:
```
homens = data.loc[data['Sexo']=='homem']
mulheres = data.loc[data['Sexo']=='mulher']
```

[17]:
```
coef_homens_idade = homens['Idade'].std()/float(homens['Idade'].mean())
coef_mulheres_idade = mulheres['Idade'].std()/float(mulheres['Idade'].mean())
print('Os coef. de variação da Idade para Homens e Mulheres são,␣
 ↪respectivamente:',coef_homens_idade,
      ' e ',coef_mulheres_idade)
```

```
Os coef. de variação da Idade para Homens e Mulheres são, respectivamente:
0.36102688463298127  e  0.3557758905612625
```

[18]:
```
coef_homens_imc = homens['Imc'].std()/float(homens['Imc'].mean())
coef_mulheres_imc = mulheres['Imc'].std()/float(mulheres['Imc'].mean())
```

[19]:
```
print('Os coef. de variação do Imc para Homens e Mulheres são, respectivamente:
 ↪',coef_homens_imc,
      ' e ',coef_mulheres_imc)
```

```
Os coef. de variação do Imc para Homens e Mulheres são, respectivamente:
0.1984425904593325  e  0.19902801346606
```

Dessa forma, podemos perceber claramente que a Idade e o Imc não devem impactar muito no Custo quando separamos homens e mulheres.

Agora que verificamos as influências das variáveis no Custo, podemos elencar as mesmas em ordem de impacto:

1ª Idade, já que o coeficiente de correlação entre Idade e Custo é o maior;

2ª Fumante, já que, quando as outras variáveis estão em equilíbrio, ainda vemos uma diferença considerável no Custo. Além disso, podemos ver no primeiro gráfico onde comparamos o Imc com o Custo de acordo com o consumo de cigarro que os fumantes pagavam mais mesmo com um Imc mais baixo;

3ª Imc, pois o coeficiente de correlação é menor que o da Idade;

4ª Número de Filhos, mas de forma bastante discreta, pois o coeficiente de correlação é bastante baixo.

A variável Região nos fornece informações duplicadas, já que verificamos que o Custo aumenta por Região pois as pessoas desta Região tem problemas de saúde, o que acarreta no aumento do Custo.

A variável Sexo não influenciou o Custo, como pudemos ver graficamente.

Vamos agora tentar prever novos valores de Custo com algoritmos de Regressão. Primeiramente, iremos testar a precisão de algoritmos de regressão linear. Se esta não for satisfatória, iremos experimentar outros algoritmos de regressão que são mais robustos. Como esse Dataset é pequeno (1338 linhas e 7 colunas), não precisamos nos preocupar em encontrar as variáveis mais relevantes por métodos de feature selection como $\chi^2$, f_classif ou PCA. Basta transformarmos as variáveis qualitativas em variáveis dummies utilizando o pacote pandas e começar o processo de testagem de modelos e seus parâmetros. A medida de precisão será calculada pelo coeficiente de determinação $R^2$.

[23]:
```
data_novo = pd.get_dummies(data,drop_first=True)
x = data_novo.drop('Custo',axis=1)
y = data_novo['Custo']
```

[24]:
```
def validacao_cruzada(a,b):
    from sklearn.model_selection import cross_val_score
    from sklearn.model_selection import KFold
    from sklearn.linear_model import LinearRegression
    from sklearn.linear_model import Ridge
    from sklearn.linear_model import Lasso
    from sklearn.linear_model import ElasticNet
    x = a
    y = b

    linear = LinearRegression()
    ridge = Ridge()
    lasso = Lasso()
    elastic = ElasticNet()

    kfold = KFold(n_splits=10)


    resultados = []
    linear_resultado = cross_val_score(linear,x,y,cv=kfold)
    linear_resultado = linear_resultado.mean()

    ridge_resultado =  cross_val_score(ridge,x,y,cv=kfold)
    ridge_resultado = ridge_resultado.mean()

    lasso_resultado = cross_val_score(lasso,x,y,cv=kfold)
    lasso_resultado = lasso_resultado.mean()

    elastic_resultado = cross_val_score(elastic,x,y,cv=kfold)
    elastic_resultado = elastic_resultado.mean()

    resultados =␣
↪[linear_resultado,ridge_resultado,lasso_resultado,elastic_resultado]
```

```
    resultados.sort()
    maximo = resultados[-1]
    dicionario = {linear_resultado:'Regressão Linear',ridge_resultado:
↪'Ridge',lasso_resultado:'Lasso',
                  elastic_resultado:'ElasticNet'}

    print('O melhor resultado foi ',maximo,' que pertence ao␣
↪modelo',dicionario[maximo])
```

[25]:
```
validacao_cruzada(x,y)
```

```
O melhor resultado foi  0.7445208159276375  que pertence ao modelo Ridge
```

[38]:
```
def melhores_parametros(a,b):
    from sklearn.model_selection import GridSearchCV
    import numpy as np
    from sklearn.linear_model import Ridge
    from sklearn.linear_model import LinearRegression
    from sklearn.linear_model import Lasso
    from sklearn.linear_model import ElasticNet
    x = a
    y = b

    linear = LinearRegression()
    ridge = Ridge()
    lasso = Lasso()
    elastic = ElasticNet()

    par_rl = {'alpha':np.array([0.05,0.5,1.0,2.0,3.0,4.0,5.0,10.0,15.
↪0,20,30,40,50])}
    par_elastic = {'alpha':np.array([0.05,0.5,1.0,2.0,3.0,4.0,5.0,10.0,15.
↪0,20,30,40,50]),
                   'l1_ratio':np.array([0,0.1,0.2,0.3,0.4,0.5,0.6,0,7,0,8,0,9,1.
↪0])}

    procura_ridge =␣
↪GridSearchCV(estimator=ridge,param_grid=par_rl,n_jobs=-1,cv=5)
    procura_ridge.fit(x,y)

    procura_lasso =␣
↪GridSearchCV(estimator=lasso,param_grid=par_rl,n_jobs=-1,cv=5)
    procura_lasso.fit(x,y)

    procura_elastic =␣
↪GridSearchCV(estimator=elastic,param_grid=par_elastic,n_jobs=-1,cv=5)
    procura_elastic.fit(x,y)
```

```
    print('Precisão do modelo Ridge:',procura_ridge.best_score_)
    print('Melhor valor de alpha para Ridge:',procura_ridge.best_estimator_.
↪alpha)
    print('')
    print('Precisão do modelo Lasso',procura_lasso.best_score_)
    print('Melhor valor de alpha para Lasso:',procura_lasso.best_estimator_.
↪alpha)
    print('')
    print('Precisão do modelo Elastic',procura_elastic.best_score_)
    print('Melhor valor de alpha para Elastic:',procura_elastic.best_estimator_.
↪alpha)
    print('Melhor l1_ratio para Elastic:',procura_elastic.best_estimator_.
↪l1_ratio)
```

[39]:
```
melhores_parametros(x,y)
```

```
Precisão do modelo Ridge: 0.7468697023944313
Melhor valor de alpha para Ridge: 0.5

Precisão do modelo Lasso 0.7470093075403292
Melhor valor de alpha para Lasso: 30.0

Precisão do modelo Elastic 0.7470093075403292
Melhor valor de alpha para Elastic: 30.0
Melhor l1_ratio para Elastic: 1.0
```

Podemos ver que as precisões dos modelos não mudam muito de um para o outro, mesmo quando alteramos os parâmetros e os combinamos de formas diferentes com o GridSearchCv. Se a precisão alcançada de, em média, 74,7% não for satisfatória, é recomendado que sejam testados modelos mais robustos como os que utilizamo o método Boosting (AdaBoostRegressor e GradientBoostingRegressor, por exemplo). Por curiosidade, vamos testar o GradientBoostingRegressor, pois este tem como base o Gradiente Descendente que é a mesma base dos modelos de Regressão Linear aqui testados.

[41]:
```
from sklearn.model_selection import KFold
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.model_selection import cross_val_score

boost = GradientBoostingRegressor()
kfold = KFold(n_splits=5)
resultado = cross_val_score(estimator=boost,X=x,y=y,cv=kfold)
resultado = resultado.mean()
print(resultado)
```

```
0.8549457452492263
```

[44]:
```
from sklearn.model_selection import GridSearchCV
import numpy as np
boost_parametros = {'learning_rate':np.array([0,1,0.2,0.3,0.4,0.5]),
        'n_estimators':np.array([50,100,150,200,250,300,400,500])}
procura_boost =␣
 ↪GridSearchCV(estimator=boost,param_grid=boost_parametros,cv=5,n_jobs=-1)
procura_boost.fit(x,y)
```

[44]:
```
GridSearchCV(cv=5, estimator=GradientBoostingRegressor(), n_jobs=-1,
             param_grid={'learning_rate': array([0. , 1. , 0.2, 0.3, 0.4, 0.5]),
                         'n_estimators': array([ 50, 100, 150, 200, 250, 300,
400, 500])})
```

[45]:
```
print('Precisão Gradient Boosting:',procura_boost.best_score_)
print('Melhor taxa de aprendizado:',procura_boost.best_estimator_.learning_rate)
print('Melhor número de estimadores:',procura_boost.best_estimator_.
 ↪n_estimators)
```

```
Precisão Gradient Boosting: 0.8535793875913864
Melhor taxa de aprendizado: 0.2
Melhor número de estimadores: 50
```

Logo, mesmo com os parâmetros default, o algoritmo de GradientBoostingRegressor tem uma precisão muito maior do que a obtida pelos algoritmos de Regressão Linear. Note que, como o Dataset possui um número baixo de amostras e variáveis, essa precisão pode ser aumentada consideravelmente, mas assim estaremos arriscando a generalidade do modelo. Queremos que nosso modelo seja generalizável, ou seja, que ele possa nor fornecer uma resposta precisa para diferentes valores de input. Então uma precisão excepcional com os dados que já obtemos não é ideal.

Este é o fim desse pequeno projeto.

[ ]: