
MACHINE LEARNING
EJ ANALYTICS

AGENDA

01 Introdução

02 *Machine learning*

03 Modelos de Classificação

04 Modelos Preditivos

05 Não supervisionado

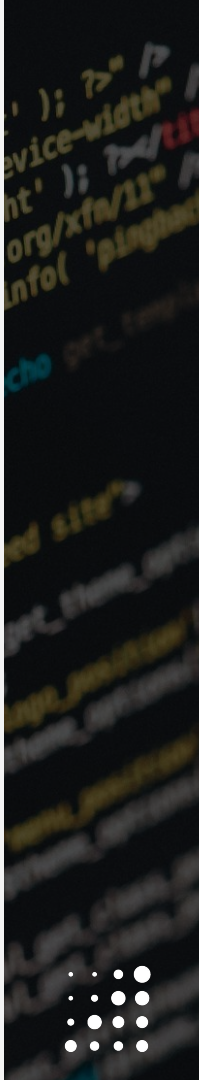
1. Introdução

01 Introdução

Esse **treinamento** busca ensinar principalmente **formas de classificar e predizer dados** por meios de modelos preditivos e de classificação. Nesse sentido, encaramos aqui um ponto central da ciência de dados, que é a **inteligência artificial (AI)**.

Inteligência artificial são mecanismos computacionais que se baseiam no **comportamento humano** para executar **problemas**, ou seja, o computador imita um **ser humano** para resolver algum problema em específico. Um dos elementos da inteligência artificial é o **aprendizado de máquina** (*machine learning)*.

Aprendizado de máquina é um sistema que pode modificar seu **comportamento automaticamente** com base na própria **experiência** que ele estiver. E, aqui, podemos pensar em **modelos mais complexos**, mas também em modelos mais simples, como um simples If do Excel.



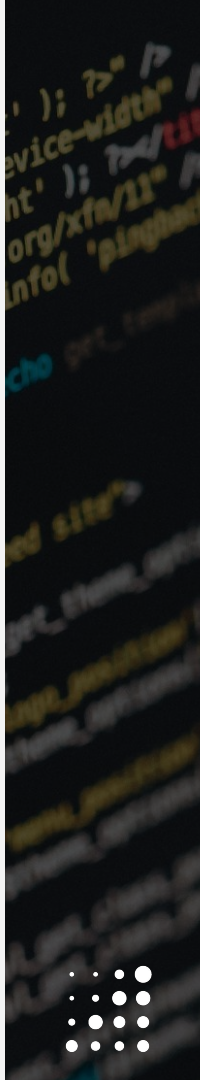
2. Machine Learning

02 *Machine learning*

Os **modelos de classificação** são um ramo dentro de *machine learning* que busca **classificar e priorizar diversas características** de um sistema de dados, de modo a associar um conjunto de informações a uma mesma **caracterização**. *Machine learning* é um **método de criação de modelos** para realizar alguma tarefa **automaticamente** e ele pode ser separado em 3 vertentes:

Aprendizado por reforço: é o famoso **tentativa e erro**, em que se envia uma informação para a máquina e ela retorna a informação do resultado, muito usado em jogos.

Aprendizado supervisionado: essa forma de AI é baseada nas **observações de resultados anteriores** para fazer previsões e esse pode ser dividido em **modelos de classificação e modelos de regressão**. Essa é uma das formas mais utilizadas em *machine learning*.

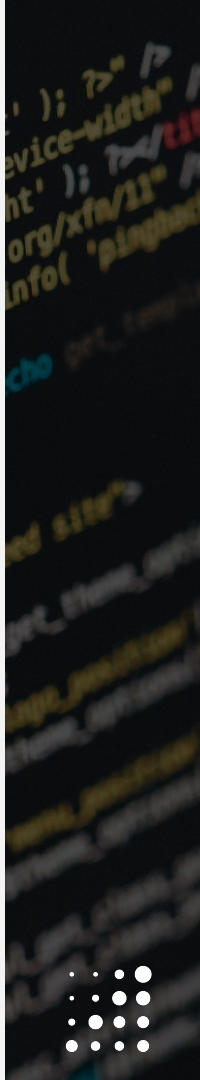


02 *Machine learning*

Aprendizado não supervisionado: essa forma de aprendizado **utiliza técnicas estatísticas para verificar padrões nos dados** e, como não há rótulos ou padrões, há outras técnicas para realizar as **predições**. Aqui existe a famosa **clusterização** dos indivíduos com base em padrões observados.

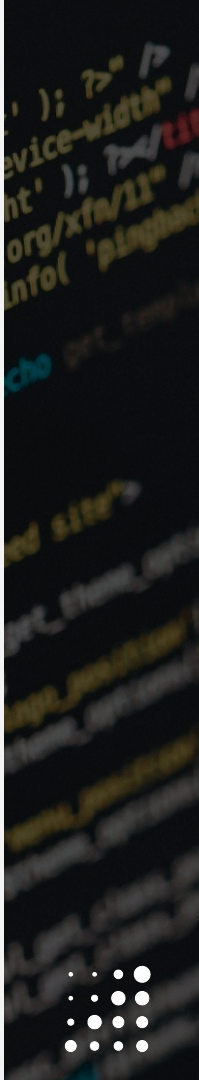
Aqui, buscamos duas ferramentas de **aprendizado supervisionado: modelos preditivos e de classificação** e, mas também buscamos entender a **clusterização**, que funciona como método de **segmentar diferentes dados**. Enquanto o modelo de predição busca determinar uma das características caso sejam observadas outras classes, o modelo de classificação busca priorizar diferentes variáveis.

No **Python**, a principal **biblioteca** que lida com *machine learning* é o **SciKit Learn**, que será muito utilizado nesse treinamento. Alguns dos seus modelos são **regressões, SVM e *Random Forest***.



02 *Machine learning*

Tarefa	Supervisionado	Não Supervisionado
Classificação	✓	
Regressão	✓	
Modelo Causal	✓	
Similarity Matching	✓	✓
Redução de dados	✓	✓
Clusterização		✓
Co-ocorrência		✓



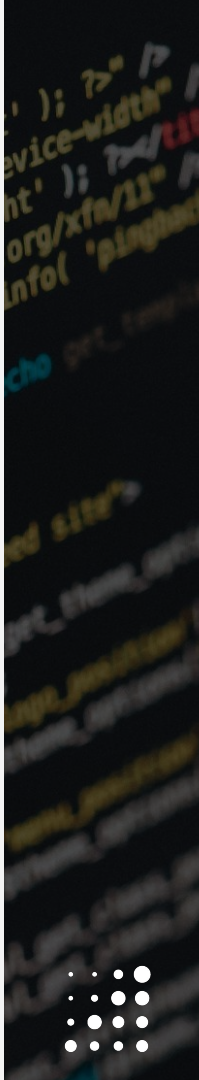
3. Modelos de Classificação

03 Classificação

No aprendizado supervisionado, há duas principais **subclasses de modelos**: os **preditivos (regressão)** e os de **classificação**. As principais diferenças deles são referentes aos seus objetivos:

Modelos Preditivos (regressão): têm um objetivo **numérico**, normalmente com um **valor que prediz qual será o resultado**. Ele só é possível se dados forem numéricos e busca entender a **correlação entre os dados** de forma a criar uma predição: “observando os dados anteriores, **se eu tiver tais números em cada variável, qual valor vou ter na outra variável?**”.

Modelos de Classificação: são modelos **categóricos** cujo principal objetivo é dizer algo sobre **classes**, de modo que, ao dar uma informação a um modelo, ele **classifica essa informação em alguma classe definida** (por exemplo, ao receber um email, ele o classifica em *spam* ou não, uma classificação **binária**), com base em outras informações já **coletadas**.



03 Classificação

Primeiramente, abordaremos **modelos de classificação**, que **lê algum *input*** e gera algum ***output*** que classifica este **input em alguma categoria**. Nesses modelos, existe uma probabilidade de atribuir aquele *input* a alguma **categoria** e esta **probabilidade é mapeada** para alguma das categorias as quais o *input* será **categorizado**.

Entretanto, é necessário pensar em qual **probabilidade vai ser o limiar** (*threshold*) **entre ser de uma categoria ou de outra** (no caso binário, que é mais simples). E, nesse sentido, as classificações do modelo podem seguir **4 características**: verdadeiro positivo, verdadeiro falso, falso positivo e falso negativo, que são mostradas em uma **matriz de confusão:**

		Detectada	
		Sim	**Não**
Real	**Sim**	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (VN)

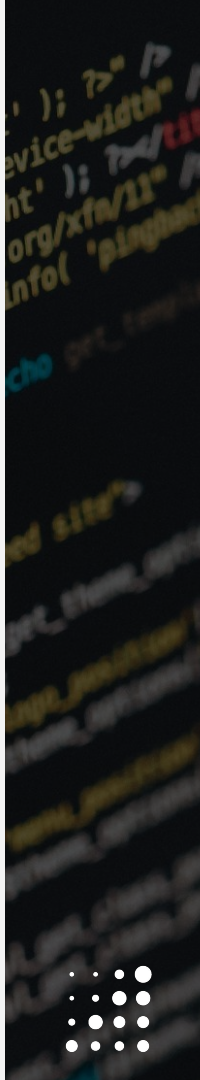
03 Classificação

Nesse sentido, há algumas formas de pensar no modelo e avaliá-lo:

Sensibilidade (ou *recall*): é a **taxa de verdadeiros positivos** em relação à **quantidade de todos os valores realmente positivos**, normalmente é uma métrica de quando você o maior número de pessoas **medidas certamente** sem tantos problemas em medir alguém erradamente.

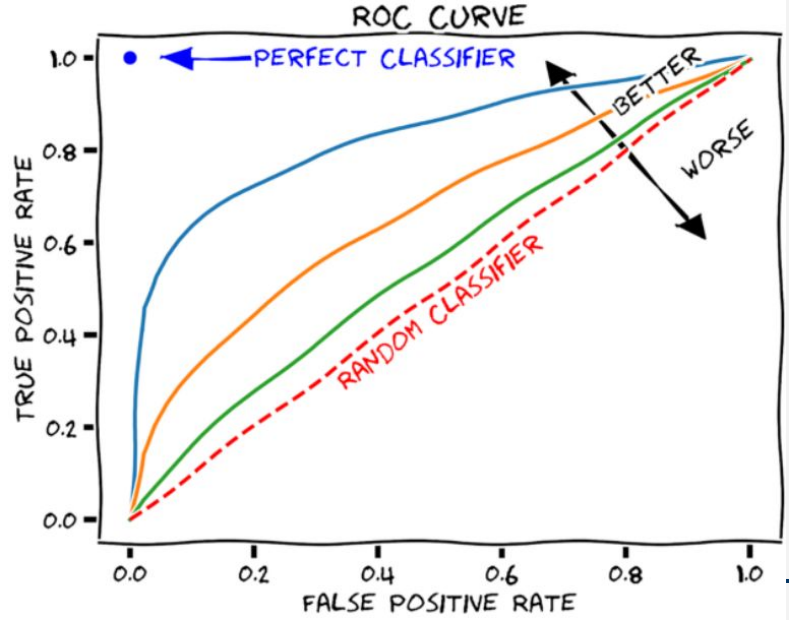
Especificidade: é a **taxa de verdadeiros negativos** em relação à **quantidade total de valores negativos**, normalmente medida quando **não se pode se dar ao luxo de errar** na mensuração.

Intuitivamente, quando o *threshold* é **muito baixo**, muitos dos dados serão marcados com a **características 1** (verdadeiro), então isso **aumenta a quantidade desses valores** verdadeiros, aumentando a **sensibilidade**, enquanto abaixa a **especificidade**.



03 Classificação

Um método de estabelecer um valor correto para sensibilidade e especificidade é o ***Receiver Operator Curve* (ROC)**, que plota a **taxa de verdadeiros positivos** em relação à **taxa de falsos positivos**, como é mostrada na imagem abaixo, que mostra o *tradeoff* entre maior verdadeiro positivo e menor falso positivo.



03 Classificação

Uma forma de escolher o *threshold* é aquele que **maximiza a média geométrica entre sensibilidade e especificidade**: **π = √(Especificidade * Sensibilidade)**, porém existem diversas discussões sobre o que deve ser mais considerado em cada modelo, então o ótimo na verdade é que depende. **Outras medidas** são:

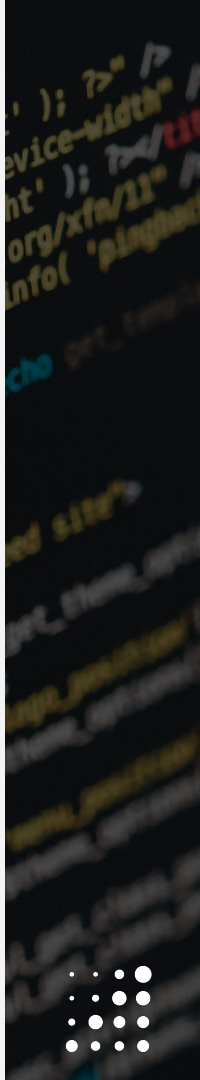
Acurácia: ela mede a **quantidade de acertos total do modelo**, então é a divisão entre a **quantidade de verdadeiros** sobre a **quantidade total** de observações. Muitas vezes ela é uma medida de **performance**. Tende a não ser boa quando a quantidade de valores negativos é muito pequena.

Precisão: parecida com a **especificidade**, é a **taxa dos verdadeiros positivos** em relação aos **todos os valores que o modelo considerou como positivos**, então mostra a quantidade de acertos do modelo em relação ao que considerou como positivo. Tende a ser boa quando os **falsos positivos são mais prejudiciais** que falsos negativos.

03 Classificação

Inclusive, em *machine learning*, também fazemos o uso de **algoritmos de validação** dos modelos, em que escolhe uma **amostra de treino (train)** e uma **amostra de teste (test)**. Essas amostras são escolhidas **aleatoriamente** e normalmente se escolhe **20%** de teste e **80%** de treino.

A amostra de **treino** é utilizada para **fitar o modelo** e a amostra de **teste** é utilizada realmente para **prever os dados** do modelo, então é um modo de usar grande parte da amostra para **encaixar o modelo** e depois **fitar** ele em uma amostra menor e independente. O sentido disso é **eliminar qualquer viés** do modelo final e gerar uma base **robusta** para interpretação. O ponto negativo disso é que nem sempre é fácil fazer com **amostras pequenas**, pois se perde muitos dados.

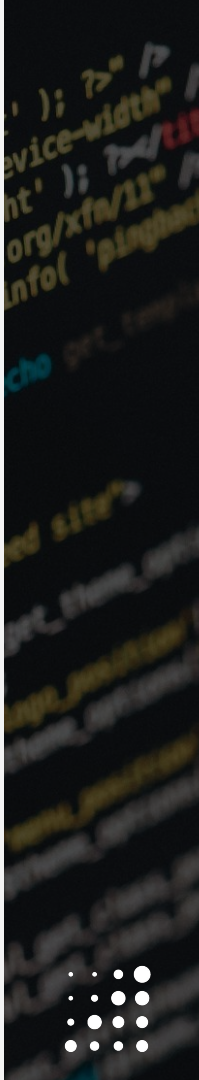


03 Classificação

São vários **tipos de modelo de classificação** e cada um tem uma certa **característica**. Para fazê-los, também foi disponibilizado um **código em Python** que roda cada um deles e depois **metrifica** qual é realmente melhor, também fazendo uma **matriz de confusão**.

***Decision Tree*:** funciona como uma **árvore** em que cada **galho** é uma **pergunta de sim e não** e, depois dessa **segregação** entre vários tipos de **características**, separa os indivíduos nesses grupos para **classificar**. É utilizado em **número e caracteres**.

***Random Forest*:** é o conjunto de várias *decision trees* que **agrega as estimativas** de diferentes **regressões** para poder **classificar** os dados. É um modelo mais **completo**, porém difícil de se **interpretar**.

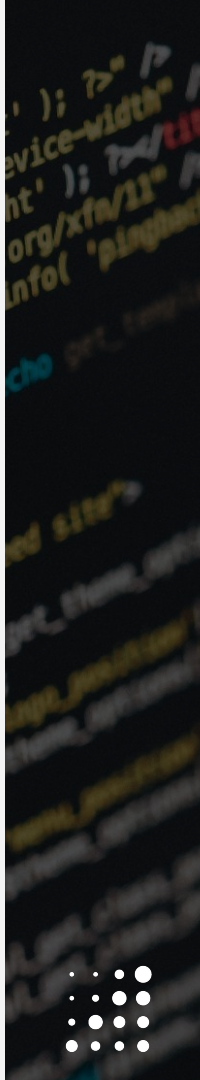


03 Classificação

Rede neural *(neural network)*: é uma forma de *machine learning* classificado como *deep learning* que **utiliza nós ou neurônios conectados e interconectados** para fazer uma estrutura como se fosse um **cérebro**. É semelhante à *decision tree*, mas muito mais **complexo** e permite **inter relações**.

Regressão Logística: classificam os **dados binariamente** quando eles **graficamente** parecem uma **sigmóide**, então muitos estão **concentrados em cada grupo** e o modelo escolhe a **probabilidade** de ele ser de **cada grupo** baseando-se na **função logística**.

Naive Bayes: é um **estimador baseado no teorema de Bayes**, considerado um estimador **rápido**, porém um estimador **ruim**, pois classifica cada variável com base nas **probabilidades**, assumindo que cada grupo é **independente**.

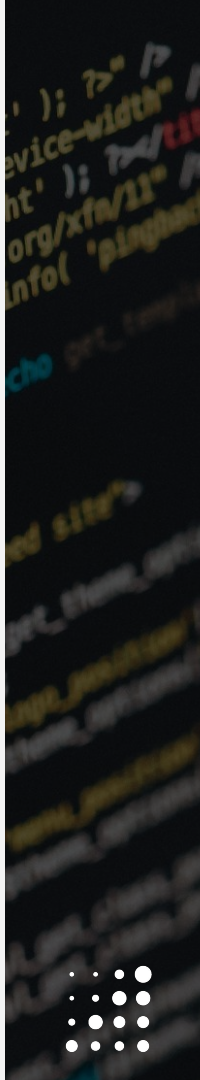


03 Classificação

K-Nearest Neighbours: ele funciona parecido com o **KMeans** no sentido de calcular a **distância de cada grupo** em relação a algum **ponto**, porém exige muito do computador e é **difícil escolher o número de K** grupos que serão feitos.

***Support Vector Machine* (SVM):** considera uma **borda** entre os grupos e a escolhe **maximizando a distância dos centros dos grupos**, o que é intuitivo **graficamente**. O *Support Vector Classifier* **(SVC)** é utilizado para **classificação** e **SVM** para dados **numéricos**.

Portanto, há **diferentes tipos de modelos** que têm qualidades e desvantagens, porém é necessário **estimar cada um** deles com base no **parâmetro** desejado (seja especificidade, sensibilidade, acurácia, precisão...) para saber qual deles deve ser **usado em cada base de dados**. Para isso, é disponibilizado também no **Python** esses **códigos** para fazer a **classificação na prática**.

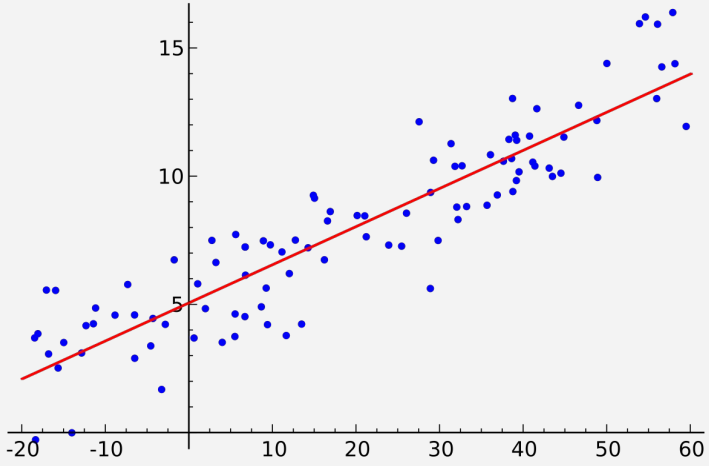


4. Modelos Preditivos

04 Predição

Além disso, também há os **modelos de regressão**, que são **formas de mensuração numérica** dos dados, de forma a entender o **efeito de várias variáveis em outra**, estabelecendo uma **previsão** de qual deve ser o **valor da variável y** (explicada ou dependente) com base nos valores das **variáveis x** (explicativa ou independente).

O modelo de **regressão linear** (a mais simples) é o seguinte:



04 Predição

A ideia da regressão é a seguinte: tentamos **explicar a variável y**, digamos **renda**, por meio de **vários outros tipos de variáveis existentes**, por exemplo anos de **escolaridade**, etnia, gênero, região, habilidade e experiência. Porém, como essa **predição não é perfeita**, há um **certo erro** também previsto na regressão.

Como é possível ver na equação a seguir, a **variável y depende de um modo linear** de cada **variável x** presente no modelo e, quando isso acontece, estamos buscando o **efeito causal** de alguma das variáveis na variável y, por exemplo o efeito da escolaridade na renda do indivíduo.

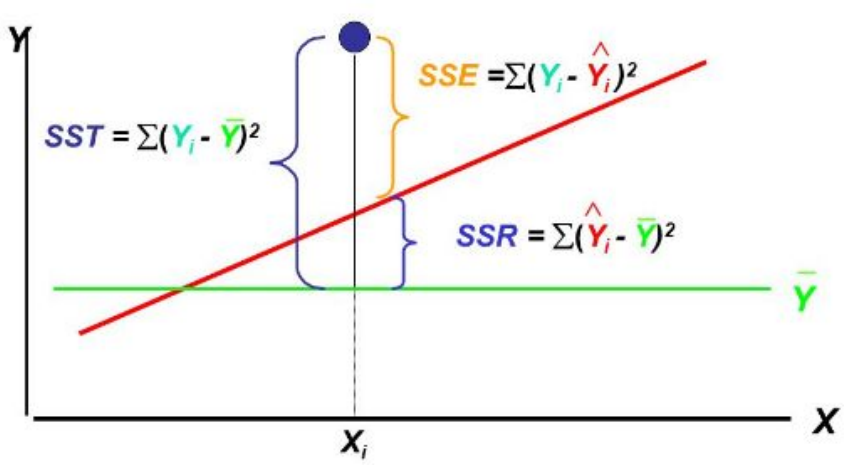
04 Predição

Depois disso, os **coeficientes** que acompanham as variáveis explicativas são os **efeitos** que desejamos calcular e, para estimá-los, há diversos métodos, mas o principal é o de **minimização da soma dos quadrados dos resíduos (SQR ou SSE)**, que gera o estimador de ***ordinary least squares*** (OLS).

Quando o modelo de regressão é estimado, a **diferença entre a variável y real** e a variável **y predita** é chamada de **resíduo**. Por isso, quando dizemos sobre o modelo geral, falamos de **erro**, porém, quando vamos **medí-lo**, estamos falando de **resíduos**, e são esses que queremos **minimizar**.

04 Predição

Quando a regressão é estimada, há **3 tipos de valores** muito importantes a serem calculados. A **soma dos quadrados totais (SQT ou SST)** é a soma das diferenças entre os **y reais de sua média** e são fixos independente da estimação do modelo. Além disso, também há a **soma dos quadrados dos resíduos** ou *squared sum of errors* **(SQR ou SSE)** que desejamos minimizar e a **soma dos quadrados explicada** ou *squared sum of regression* **(SQE ou SSR)**.



04 Predição

Quando a regressão é estimada, há **3 tipos de valores** muito importantes a serem calculados. A **soma dos quadrados totais (SQT ou SST)** é a soma das diferenças entre os **y reais de sua média** e são fixos independente da estimação do modelo. Além disso, também há a **soma dos quadrados dos resíduos** ou *squared sum of errors* **(SQR ou SSE)** que desejamos minimizar e a **soma dos quadrados explicada** ou *squared sum of regression* **(SQE ou SSR)**.

Sum of Squares Total → $SST = \sum (y - \bar{y})^2$

Sum of Squares Regression → $SSR = \sum (y' - \bar{y'})^2$

Sum of Squares Error → $SSE = \sum (y - y')^2$

$$\mathrm{SST_{otal}} = \mathrm{SSE_{xplained}} + \mathrm{SSR_{esidual}}$$

or, equivalently,

$$\sum (y_i - \bar{y})^2 = \sum (\hat{y_i} - \bar{y})^2 + \sum (y_i - \hat{y_i})^2$$

and

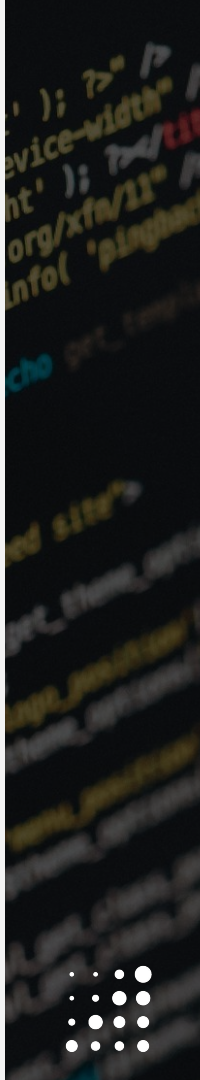
$$R^2 = 1 - \frac{\mathrm{SSR_{esidual}}}{\mathrm{SST_{otal}}}$$

04 Predição

E, por fim, também temos o **R^2**, que expressa o quanto a **regressão fita bem os dados** ou não e é um valor de **0% a 100%**. Quanto mais próximo o valor do **R^2 é de 1,** melhor o modelo explica a **variável y**, porém nem sempre um **R^2 muito grande pode ser positivo**.

Em geral, há um grande problema quando existem variáveis demais no modelo, o que pode ser chamado de ***overfitting***, o que pode causar um **R^2 grande** pois y está muito **explicada**. O principal problema disso é a **multicolinearidade**, uma situação em que as **variáveis** presentes **em x estão correlacionadas entre si**, o que gera um problema de **variância** (a variância do estimador de beta é grande), por isso, é importante fazer uma **tabela de correlações** para nunca colocar variáveis muito correlacionadas juntas.

O principal problema de ter uma **variância alta** é que a precisão de acerto vai ser **menor**, pois o valor de **beta** varia muito e, com isso, os **testes de hipóteses perdem sua força**.



04 Predição

Uma **tabela de correlação** funciona da seguinte forma: ela mostra a **correlação entre duas variáveis** presentes na base de dados. Por isso, a **diagonal principal** dessa tabela é totalmente igual a **1** já que a correlação de uma variável com ela mesma é **100%**. Além disso, essa tabela é **simétrica**, pois a correlação entre duas variáveis não é determinada pela posição delas.

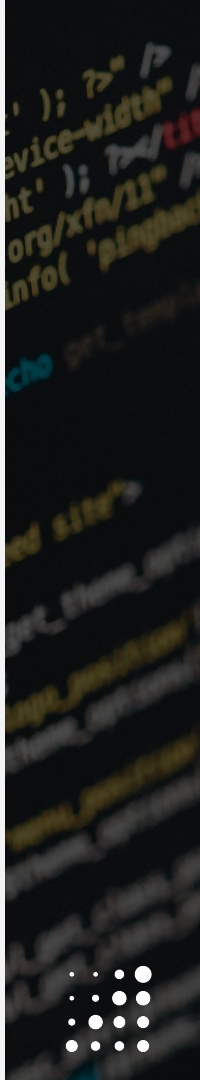
	preg	plas	pres	skin	test	mass	pedi	age	class
preg	1.000000	0.129459	0.141282	-0.081672	-0.073535	0.017683	-0.033523	0.544341	0.221898
plas	0.129459	1.000000	0.152590	0.057328	0.331357	0.221071	0.137337	0.263514	0.466581
pres	0.141282	0.152590	1.000000	0.207371	0.088933	0.281805	0.041265	0.239528	0.065068
skin	-0.081672	0.057328	0.207371	1.000000	0.436783	0.392573	0.183928	-0.113970	0.074752
test	-0.073535	0.331357	0.088933	0.436783	1.000000	0.197859	0.185071	-0.042163	0.130548
mass	0.017683	0.221071	0.281805	0.392573	0.197859	1.000000	0.140647	0.036242	0.292695
pedi	-0.033523	0.137337	0.041265	0.183928	0.185071	0.140647	1.000000	0.033561	0.173844
age	0.544341	0.263514	0.239528	-0.113970	-0.042163	0.036242	0.033561	1.000000	0.238356
class	0.221898	0.466581	0.065068	0.074752	0.130548	0.292695	0.173844	0.238356	1.000000

04 Predição

Nessa mesma linha, também é possível fazer **testes de hipóteses** dentro de uma **regressão**. É possível realizar **testes t** para testar se cada **coeficiente é igual a zero** e, nesse caso, analisa-se o **p-valor** de cada coeficiente para saber se ele é ou não importante a ser considerado.

Caso o **p-valor seja maior que 5%**, aceitamos a hipótese nula, o que significa que aquela variável **não tem impacto significativo** em y e portanto pode ser **desconsiderada**. Em um projeto, isso significa que essa variável **não é importante na escolha de maximizar** a renda ou receita da empresa.

Além disso, também há um **teste f** que testa se **todos os coeficientes juntos são iguais a zero**, para verificar se a regressão é ou não útil nesse caso e, se seu **p-valor** for maior do que 5%, a **regressão não tem importância significativa**.



04 Predição

Um exemplo de *output* de uma **regressão** é a seguinte **tabela**, que diz que há **614 observações**, tem **R^2 de 30%** e apenas as variáveis **“skin”** e **“test”** são **não significativas**, ou seja podem ser desconsideradas pelo cliente. Além disso, também há intervalos de confiança para **erro tipo 1** de **5%** para analisar se o coef está dentro ou não do intervalo.

OLS Regression Results

Dep. Variable:	class	R-squared:	0.304
Model:	OLS	Adj. R-squared:	0.295
Method:	Least Squares	F-statistic:	33.07
Date:	Thu, 08 Sep 2022	Prob (F-statistic):	3.01e-43
Time:	06:23:32	Log-Likelihood:	-307.96
No. Observations:	614	AIC:	633.9
Df Residuals:	605	BIC:	673.7
Df Model:	8		
Covariance Type:	nonrobust		

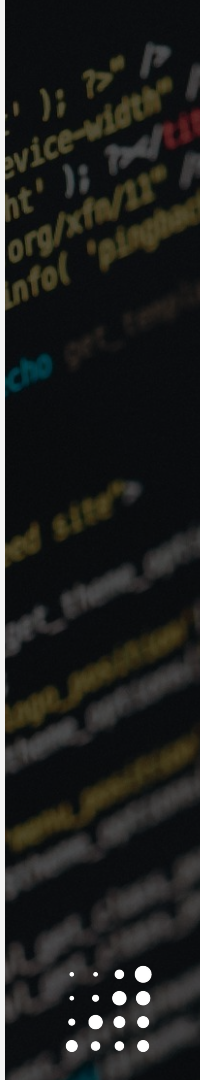
	coef	std err	t	P>|t|	[0.025	0.975]
const	-0.8433	0.098	-8.601	0.000	-1.036	-0.651
preg	0.0210	0.006	3.676	0.000	0.010	0.032
plas	0.0055	0.001	9.677	0.000	0.004	0.007
pres	-0.0027	0.001	-2.852	0.004	-0.005	-0.001
skin	0.0007	0.001	0.608	0.543	-0.002	0.003
test	-0.0003	0.000	-1.603	0.109	-0.001	5.84e-05
mass	0.0140	0.002	6.019	0.000	0.009	0.019
pedi	0.1452	0.050	2.891	0.004	0.047	0.244
age	0.0039	0.002	2.218	0.027	0.000	0.007

04 Predição

Outro tipo de análise a ser feita na **regressão** é o valor da **regressão calculado pelo coeficiente**. Nesse caso, podemos observar o valor do coeficiente para saber qual **variável tem maior impacto sobre a variável y** e, por isso, deve ser **priorizada** pelo cliente. Porém, como normalmente as variáveis x são diferentes, é importante **padronizá-las dividindo seu valor pelo maior valor da variável**.

Com isso, é possível realizar uma **análise de priorização das variáveis** dentro do modelo e essa é uma das formas de realizar um **modelo de priorização**. Para isso, selecionam-se quais variáveis realmente importam para o cliente para não ocorrer o *overfitting* também e fazer os **testes de hipóteses** serem pouco importantes.

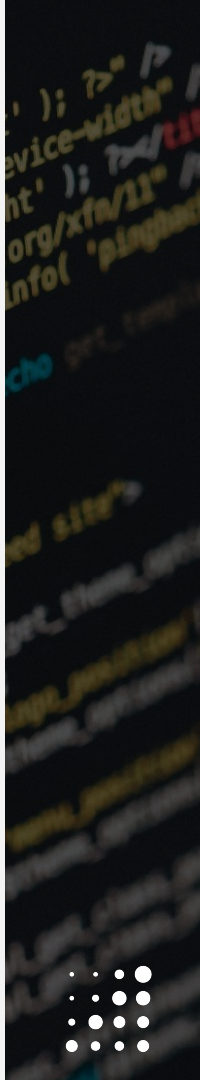
Também é possível **realizar predições** nesses modelos, já que, **com vários x** calculados, é possível **determinar o y predito** com base nos coeficientes.



04 Predição

Por exemplo, no caso anterior da tabela, **buscamos saber o efeito de cada variável na determinação se há ou não câncer**. Outras situações é **regredimos a receita de firmas** (caso o desejo seja aumentar a receita) em **várias variáveis que a impactam**, por exemplo *marketing*, estado de atuação (*dummies* para cada estado), setor (também *dummies*), investimento em maquinário, investimento em imobilizado e investimento em qualidade.

Nesse caso, olhamos primeiro para o **p-valor** de cada variável para verificar se ela é **significativa**, pois, se não for, já é possível d**escartá-la da análise**. Agora, em relação às **variáveis significantes** (as que possuem **p-valor menor que 5%** e têm ***), olhamos também para os seus coeficientes. O significado do coeficiente é que o **aumento de 1 unidade em x aumenta y** na quantidade do coeficiente. Por exemplo, se o coeficiente de *marketing* for 3 e de maquinário for 5, então é mais **eficiente** para a empresa investir em maquinário nesse momento, então é possível fazer um ***ranking*** das categorias da **regressão**, mas deve-se tomar muito cuidado com o que coloca em x.

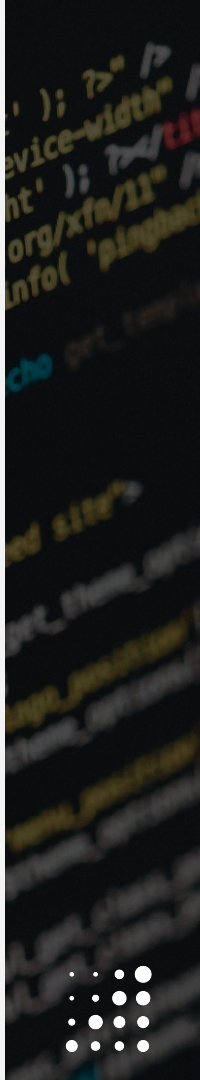


04 Predição

Além desse **tipo de regressão mais clássico**, também há as regressões em que a variável **x é o tempo**, ou seja, a variável y tem um certo **padrão de crescimento** conforme o tempo e pode ser predita para o futuro com base na sua história. O modelo mais simples disso é **baseado no tempo** e pode ser de diferentes tipos:

Linear: esse é o **modelo mais simples** e mais utilizado, em que a variável y depende **linearmente** das outras, de modo que um aumento de 5 em x aumenta y na mesma proporção.

Exponencial: esse é um **modelo de crescimento** em que a variável y **cresce** mais **rapidamente** ou mais **decresce** mais **lentamente** conforme maiores as observações de x e ele funciona bem quando aumentos em alguma das variáveis x aumenta excessivamente o valor de y.

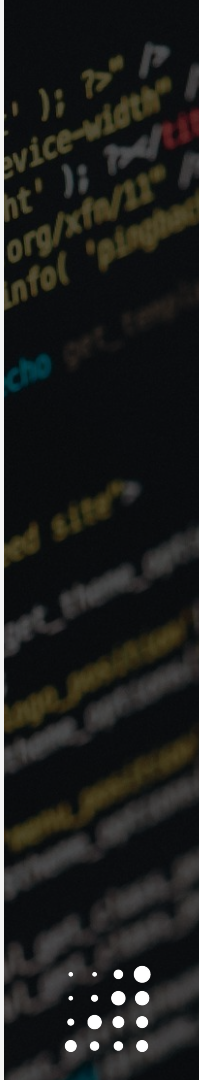


04 Predição

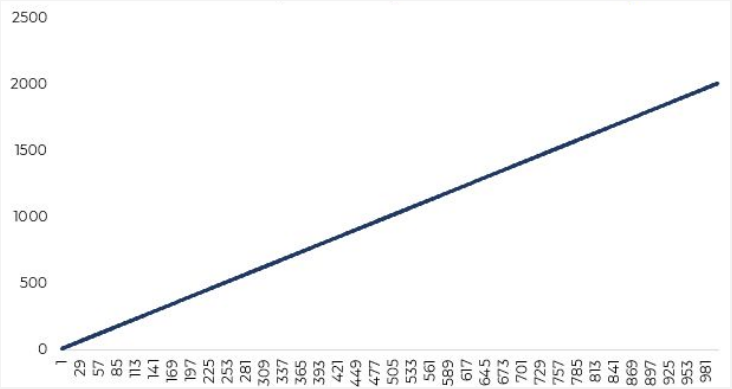
Logarítmica: esse é um modelo de crescimento percentual, em que o coeficiente que **multiplica o logaritmo** é o efeito de um **aumento de 1% de x na variável y**, além de que é um modelo que cresce em uma **tendência** menor conforme maiores valores de x.

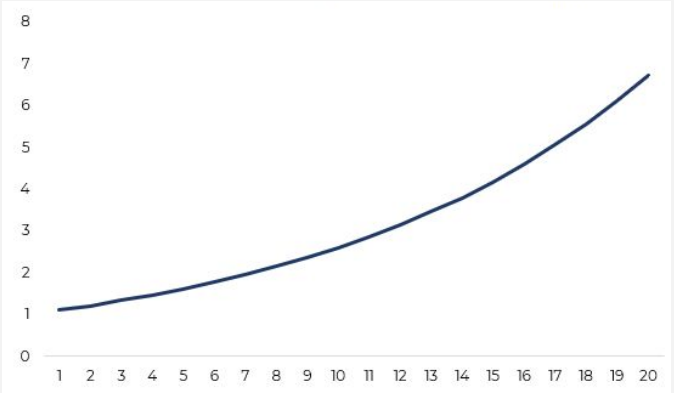
Logística: é um dos principais modelos utilizados e tem uma certa característica: ele **cresce rapidamente por um momento e depois passa a decrescer**, atingindo um **patamar** final. Ele é adquirido por meio de **1 dividido por uma exponencial**.

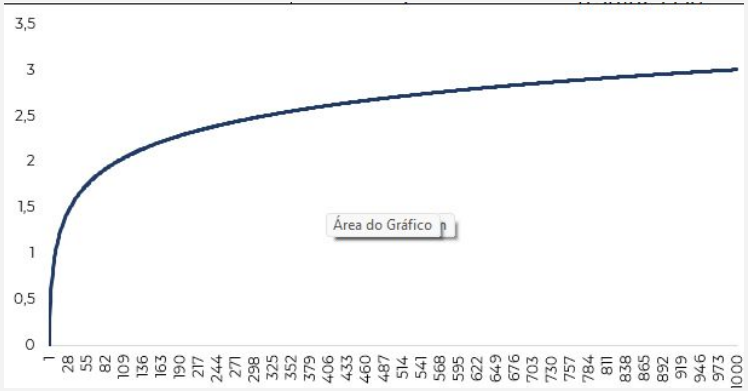
Hipérbole (1/x): esse modelo é bem parecido com uma **exponencial** de expoente **negativo**, porém com um **decrescimento menor** em relação à exponencial, o que faz com que ela seja preferível em alguns casos.

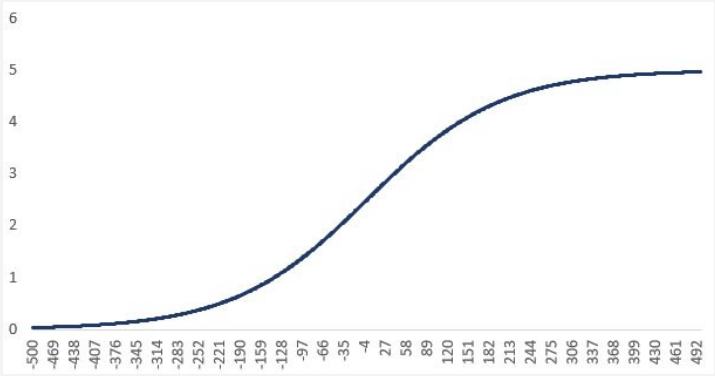


04 Predição


2500
2000
1500
1000
500
0
1 29 57 85 113 141 169 197 225 253 281 309 337 365 393 421 449 477 505 533 561 589 617 645 673 701 729 757 785 813 841 869 897 925 953 981


8
7
6
5
4
3
2
1
0
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

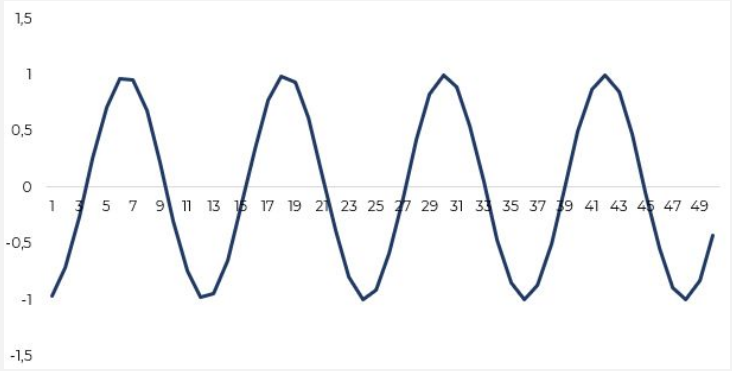

3,5
3
2,5
2
1,5
1
0,5
0
Área do Gráfico
1 28 55 82 109 136 163 190 217 244 271 298 325 352 379 406 433 460 487 514 541 568 595 622 649 676 703 730 757 784 811 838 865 892 919 946 973 1000

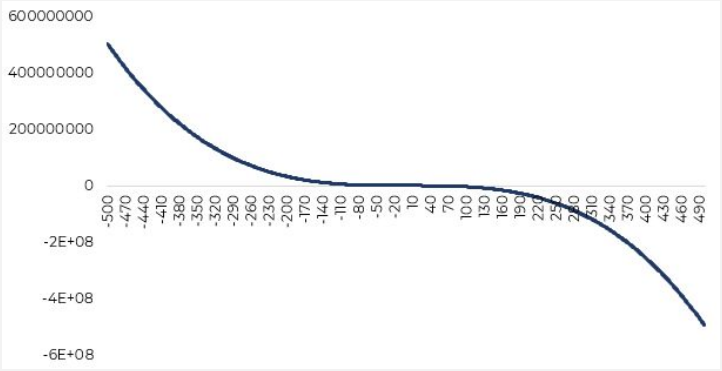

6
5
4
3
2
1
0
-500 -469 -438 -407 -376 -345 -314 -283 -252 -221 -190 -159 -128 -97 -66 -35 -4 27 58 89 120 151 182 213 244 275 306 337 368 399 430 461 492

04 Predição

Senóide: é um **modelo de oscilação** ao redor de algum valor e funciona muito bem para **variáveis sazonais** quando condicionado no tempo, o que faz muito sentido em **modelos de evolução no tempo** (em que x é o tempo).

Polinomial: esse é o **modelo mais genérico** pois todas as funções acabam se **aproximando** de alguma polinomial com algum grau, mas vale mediar qual modelo é melhor, pois nem sempre o **modelo com maior acurácia** é o **melhor**, principalmente quando utiliza muitos **parâmetros** para o estimar.

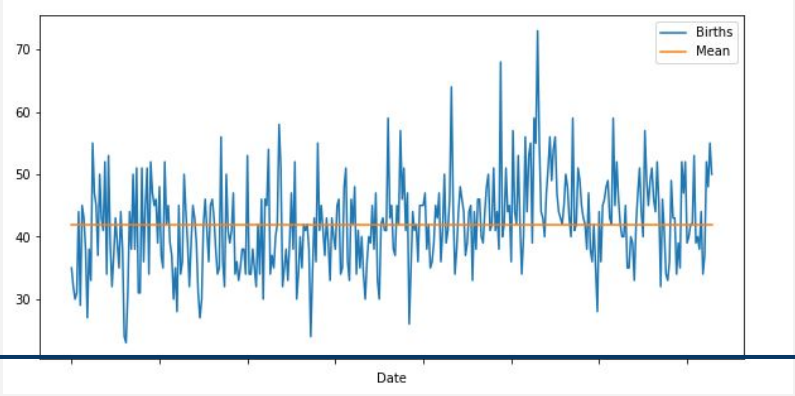




04 Predição

Esses modelos de **regressão no tempo não são os únicos** e também há muitos modelos mais **complexos** que esses. Em geral, temos alguns conceitos a serem definidos nesse caso que fogem das **séries** anteriores.

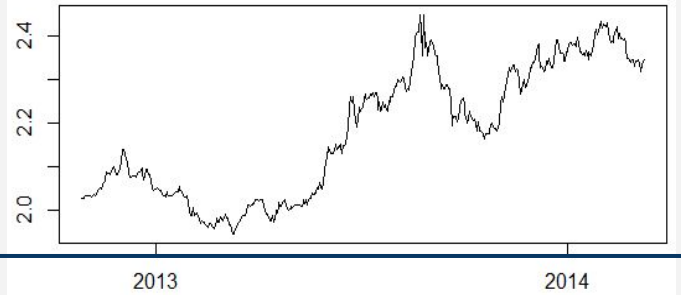
Série estacionária: é aquela série que possui **média, variância constantes** e **não possui correlação serial** (correlação com seu período anterior). Nesse caso, o período anterior da amostra **não impacta em nenhum modo o período seguinte**, gerando uma série muito parecida com a seguinte, em que o valor é **totalmente aleatório**.



04 Predição

Série com raiz unitária: ao contrário da **série estacionária**, as séries de raiz unitária têm **médias diferentes conforme o tempo** e não flutuam ao longo de um **valor fixo** (no caso a linha vermelha anterior). Nesse caso, há uma **linha de tendência**, em que a série flutua ao longo do tempo, positiva ou negativa. A mais famosa dessas séries se chama **passeio aleatório** e cada período impacta no período seguinte, então há **memória** da série.

Para **tornar** a série **estacionária**, é importante **tirar a primeira diferença**, que o valor de um **período menos** o valor de seu **período anterior**, o que gera uma nova série que é estacionária. Outro modo de fazer isso é **retirando os retornos**, dividindo o valor da série pelo valor anterior e subtraindo 1.

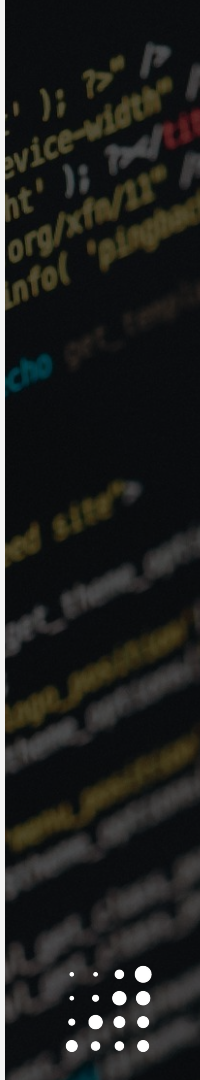


04 Predição

Além disso, também é possível **unir os dois casos**: aquele com *cross-section* **(dados de diferentes indivíduos** no único período de tempo) e **séries de tempo** (dados de um indivíduo para **diferentes períodos de tempo**) e gerar séries em **painel**.

Computacionalmente essas s**éries são fáceis também de se calcular**, mas a grande importância delas é que elas normalmente podem **melhorar a estimativa** do modelo de **predição** ao utilizar diferentes modelos, por isso são importantes.

Entretanto, como é **difícil obter dados** desse tipo para a EJ, não entraremos muito a fundo, porém é importante saber que esses **modelos existem**.



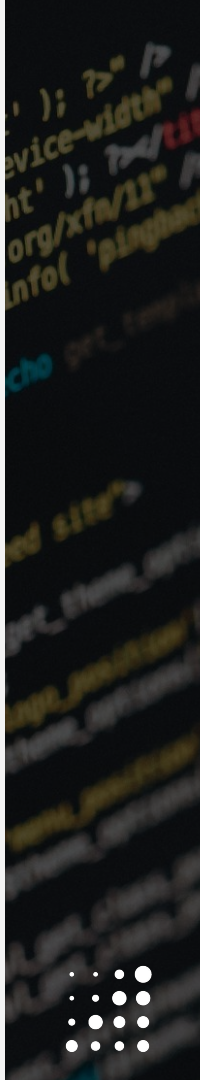
5. Não Supervisionado

05 Não supervisionado

Como já foi explicado, essa **forma de aprendizado utiliza técnicas estatísticas** para verificar **padrões** nos dados não observados ainda. Por isso, é possível utilizar modelos não supervisionados para realizar **classificações** das variáveis, porém, como esses modelos **não são preditivos**, a análise necessária deles é um **pouco diferente** da habitual.

Clusterização: é uma forma de **agregar os dados** com base nas suas **características**, sem que essa classificação seja baseada em uma **característica** já existente e, por isso, é diferente dos modelos de classificação. Esse método cria grupos (ou *clusters*) com base em características desejadas de se ter no grupo e permite **dividir a análise** para cada grupo.

Há diversos tipos de **clusterização**, porém os mais famosos são a **clusterização hierárquica** e a **clusterização KMeans**. A clusterização hierárquica não é tão mais utilizada e ela é muito parecida com o modelo de ***Decision Tree***, porém a *KMeans* ainda é muito utilizada principalmente por seu **poder de interpretação**.



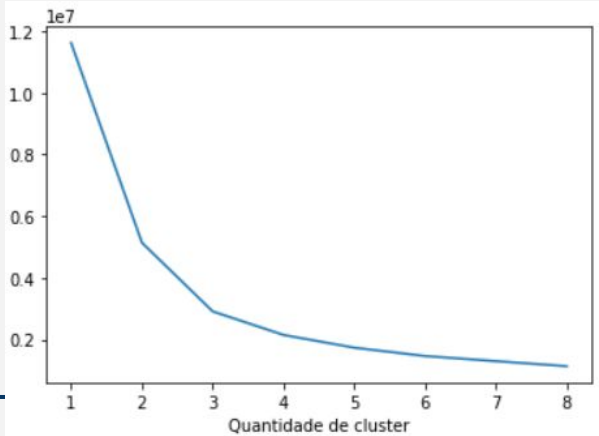
05 Não supervisionado

Como é mostrado no gráfico, a clusterização *KMeans* **separa os dados** com base em **eixos de um gráfico** e, com base na quantidade de *clusters* a ser **definida**, captura o **centróide** de cada *cluster* de forma a **minimizar a soma das distâncias ao quadrado** dos dados até esse centróide **(SSE)**. Por isso, ele permite interpretar os dados com base em **distâncias** e os dados do mesmo *cluster* estão em lugares parecidos.

Quando há mais de 2 variáveis, o gráfico fica em **n dimensões**.

05 Não supervisionado

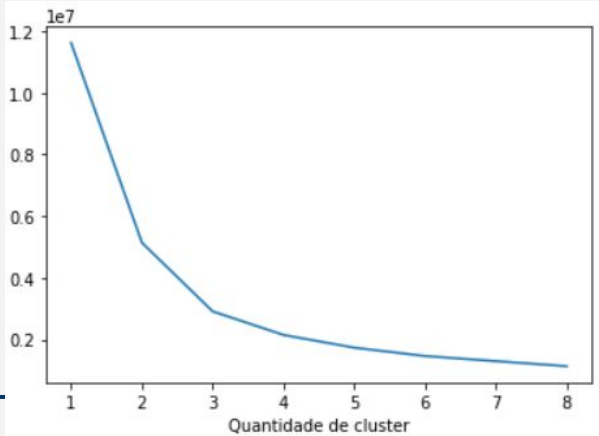
Porém, também é necessário saber o **número ideal** de *clusters* a serem feitos. Para isso, **utilizamos a soma das distâncias ao quadrado** dos dados até seu **centróide (SSE)** e a plotamos para **diferentes números de *cluster*** definidos. Logicamente, quanto mais *clusters*, menor vai ser essa distância, até o ponto do número de *clusters* ser igual ao número de dados, em que a **distância é 0**. Porém, como escolhemos o número de *clusters* ótimo?



05 Não supervisionado

Para isso, olhamos para a **taxa de redução** dessa distância ao longo do gráfico seguinte. Como é possível verificar, a **maior taxa de redução da distância** é quando passamos de 1 para 2 *clusters* e, nesse sentido, **o ótimo é selecionar 2 *clusters***, pois um número maior que esse não diminui tanto a distância.

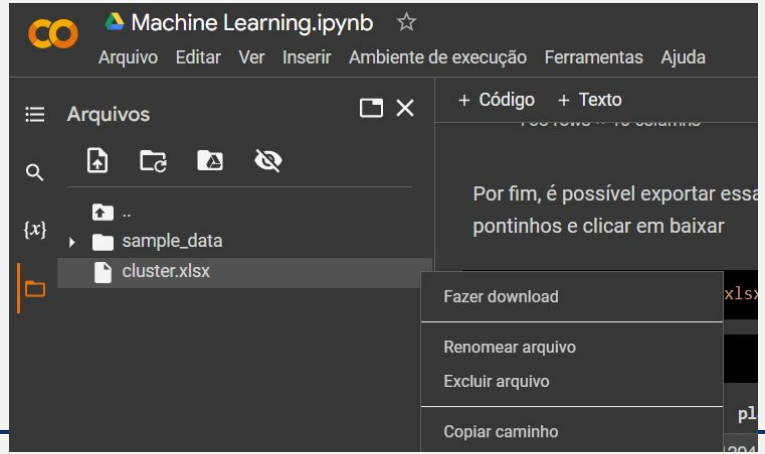
Entretanto, é possível sim escolher um número de *clusters* não ótimo, mas deve-se **verificar se isso faz sentido qualitativo** para a análise feita.



05 Não supervisionado

Além disso, também deve-se atentar se os **valores não estão muito próximos** entre si. Caso a distância não diminua muito **significativamente** ao longo do gráfico, é possível que o número de *clusters* **ótimo seja 1**, o que significa que todos os dados se comportam **semelhantemente** em um mesmo grupo, o que também é possível.

Por fim, basta **baixar a base** conforme mostra o *print* seguinte:



05 Não supervisionado

Quanto à **interpretação de cada *cluster*** feito, é possível separar cada um dos *clusters* por meio do **filtro do Excel** e **tirar suas médias para comparar cada variável** entre cada *cluster* feito. Depois disso, basta **interpretar as características** de cada *cluster* e entender qualitativamente o que cada um significa e qual significado possui para a interpretação geral do projeto. E, conforme cada projeto exige, é possível que haja **diferentes resultados** buscados com base nisso.

Cluster 2

Municipio	População	renda_pc_max_decil_9	indice_gini	prop_renda_10_
Armação dos Búzios	35060	1600	0.51	42
Petrópolis	307144	1855	0.55	46
Foz do Iguaçu	257971	1670	0.53	43
Tibau do Sul	14694	837	0.6	51
Média	**153717.25**	**1490.50**	**0.55**	**45.50**

Comparação médias

Cluster	População	renda_pc_max_decil_9	indice_gini	prop_renda_10_
Cluster 0	97,833.25	1,888.25	0.54	45.25
Cluster 1	82,111.15	1,218.00	0.53	42.92
Cluster 2	153,717.25	1,490.50	0.55	45.50

Comparação médias %

Cluster	População	renda_pc_max_decil_9	indice_gini	prop_renda_10_
Cluster 0	29.32%	41.08%	33.31%	33.85%
Cluster 1	24.61%	26.50%	32.75%	32.11%
Cluster 2	46.07%	32.43%	33.93%	34.04%

05 Não supervisionado

Uma das opções é a **realização de um *cluster* ideal**, que seria aquele *cluster* que mais **corresponde às características ideais da empresa** em questão, como se fosse uma **persona ideal** que o *marketing* foca na hora de prospectar. Outra forma é selecionar cada um dos *clusters* para **algum tipo de estratégia** ou modelo de negócio a ser seguido. E, por fim, uma estratégia válida é **ranquear os indivíduos** com base no ***cluster* ideal** da empresa, obtendo um **modelo de priorização** não supervisionado.

Cluster 2

Municipio	População	renda_pc_max_decil_9	indice_gini	prop_renda_10_
Armação dos Búzios	35060	1600	0.51	42
Petrópolis	307144	1855	0.55	46
Foz do Iguaçu	257971	1670	0.53	43
Tibau do Sul	14694	837	0.6	51
Média	**153717.25**	**1490.50**	**0.55**	**45.50**

Comparação médias

Cluster	População	renda_pc_max_decil_9	indice_gini	prop_renda_10_
Cluster 0	97,833.25	1,888.25	0.54	45.25
Cluster 1	82,111.15	1,218.00	0.53	42.92
Cluster 2	153,717.25	1,490.50	0.55	45.50

Comparação médias %

Cluster	População	renda_pc_max_decil_9	indice_gini	prop_renda_10_
Cluster 0	29.32%	41.08%	33.31%	33.85%
Cluster 1	24.61%	26.50%	32.75%	32.11%
Cluster 2	46.07%	32.43%	33.93%	34.04%

Obrigado!