

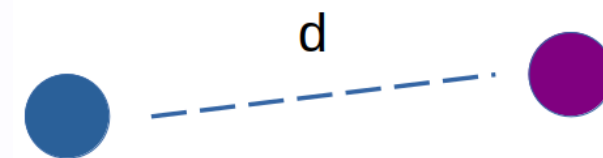
# K-nearest neighbors (KNN)

Prof. Me. Alexandre Henrick  
Sistemas de Informação - 8º P

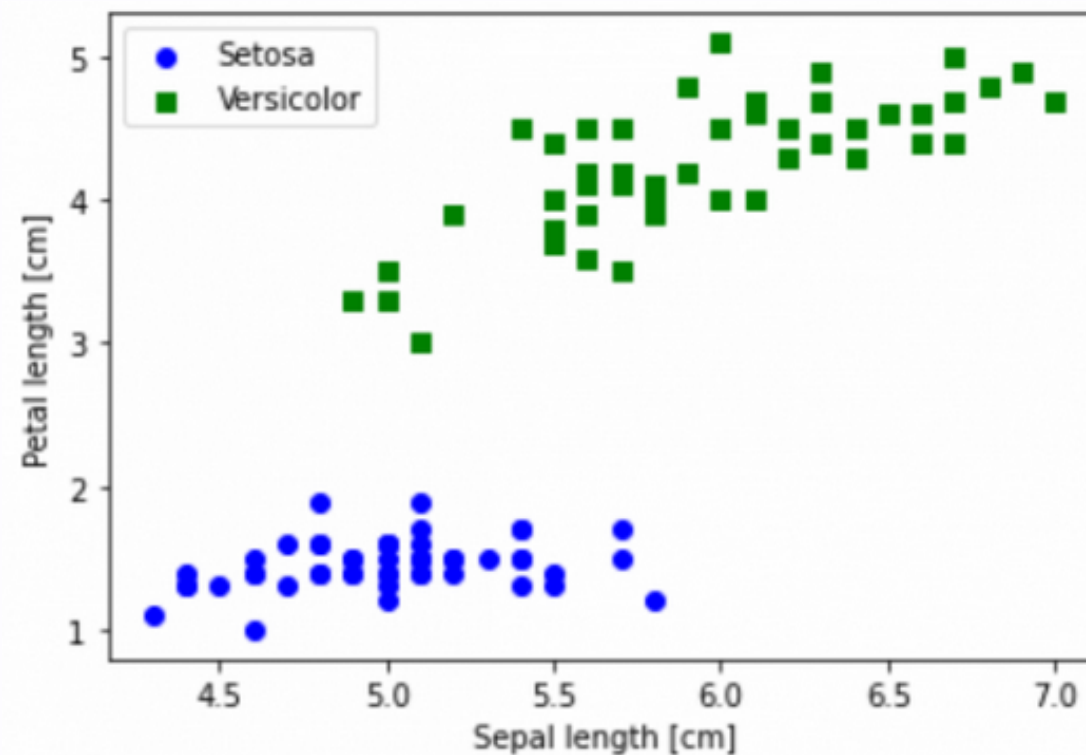
# Medidas de distância

- É uma abordagem **extremamente** popular, principalmente nos dias atuais, para criação de algoritmos de ML
- Ultimamente vem sendo bastante utilizado em técnicas que precisam processar ou classificar texto
- A ideia é medir a distância entre dois pontos no espaço amostral

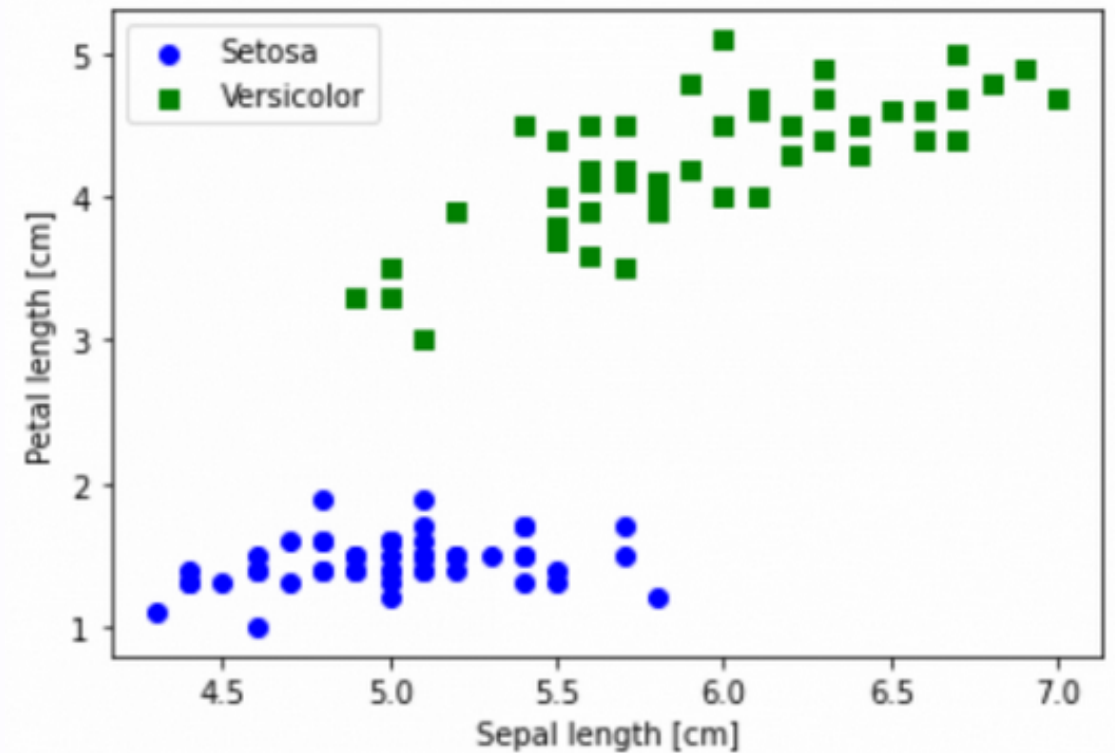
- É a medida da separação de dois pontos
- Geralmente representado por vetores no espaço amostral



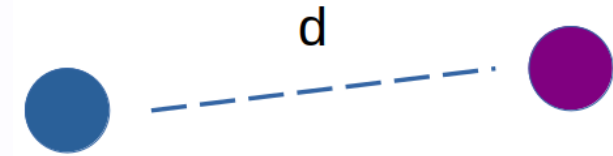
- O plot ao lado representa as observações do Iris Dataset. Estão representadas pelas variáveis **Petal Length** e **Sepal Length**



- Cada ponto pode ser representado por vetores como:  $[2, 4.5]$ ,  $[4, 6.0]$ ,  $[5.5, 3]$

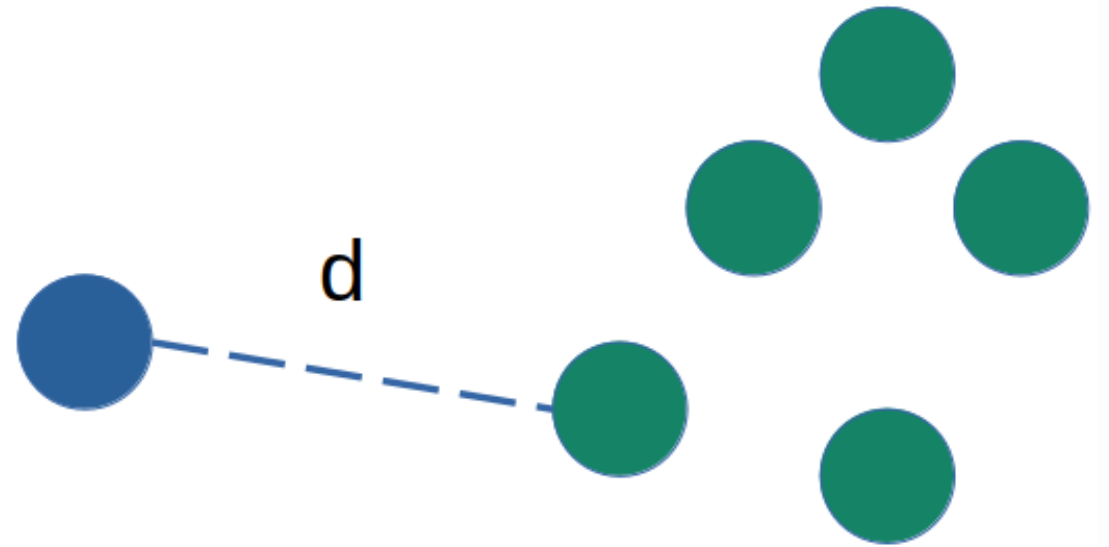


- Em machine learning, podemos usar a distância para representar a **similaridade** entre dois objetos

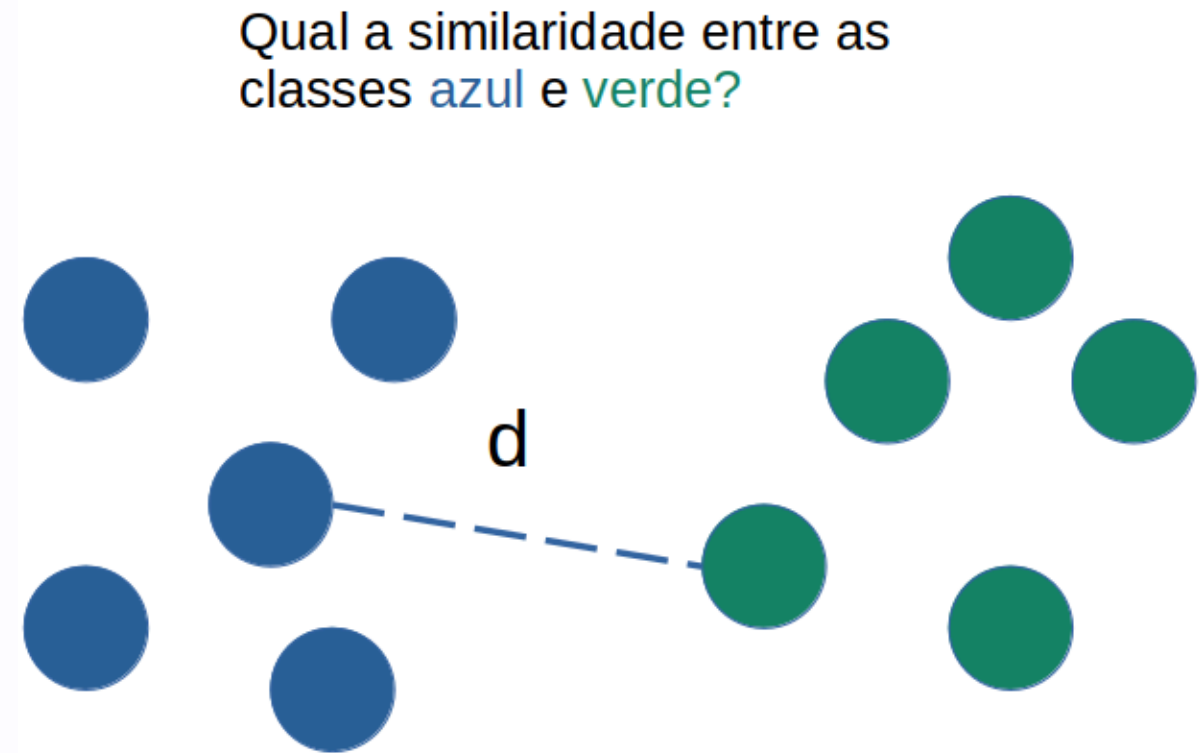


- Ao identificar **grupos** nos nossos conjuntos de dados, podemos determinar a similaridade de um ponto entre classes

Qual a similaridade entre a amostra azul e a classe verde?

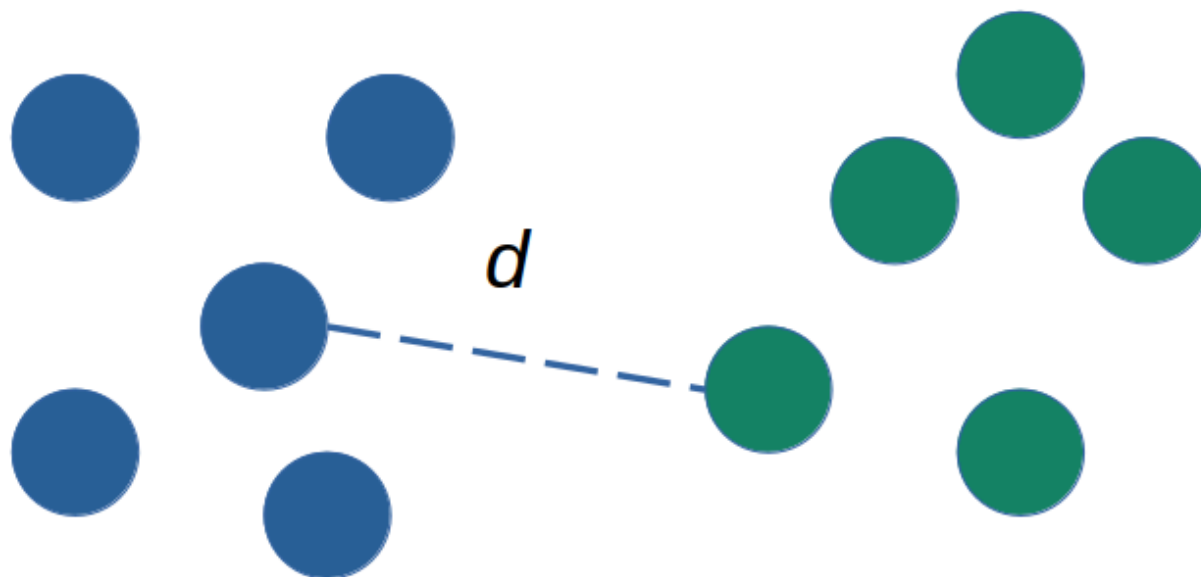


- Ou ainda calcular a similaridade **entre classes**



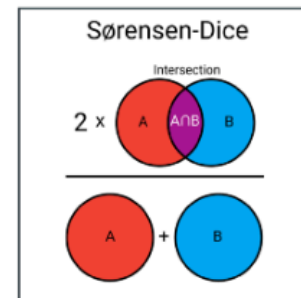
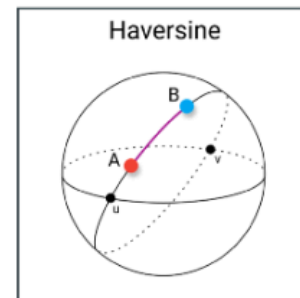
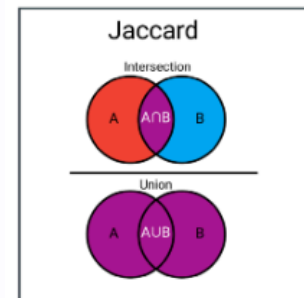
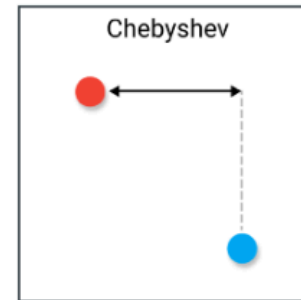
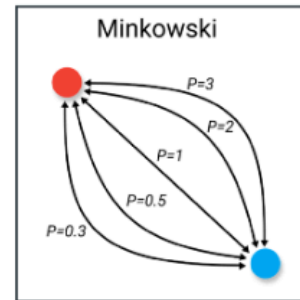
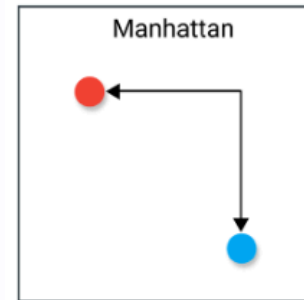
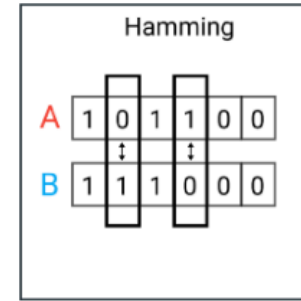
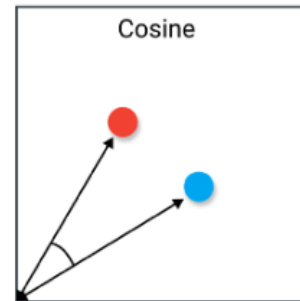
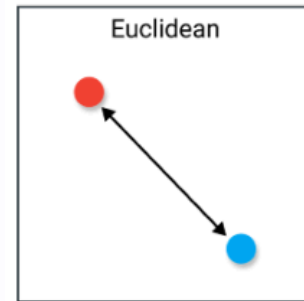


Como definir essa distância  $d$ ?



# Medidas de distância

- Para definir uma distância, **usamos métricas**
- É uma formalização do conceito de distância
- Para uma função ser considerada uma distância, deve seguir 3 axiomas:
  - $d(x, y) = d(y, x)$ , simetria
  - $d(x, y) \geq 0$
  - $d(x, x) = 0$



# Distância Euclidiana

- É a distância mais comum entre dois vetores  $p, q$
- Aquela mesma distância medida utilizando uma régua. Baseado no Teorema de Pitágoras
- Elevamos ao quadrado para garantir as diferenças positivas
- E tiramos a raiz para voltar a escala de distância normal

$$d(p, q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2}$$

# KNN

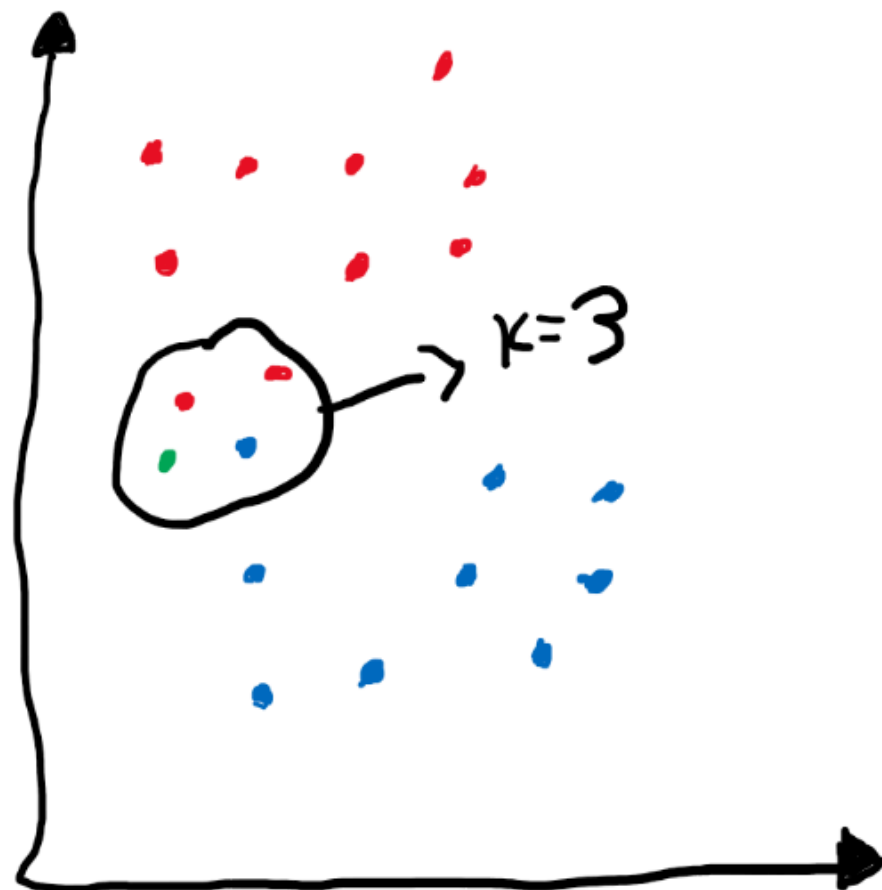
- Conhecido também como K-vizinhos mais próximos
- Algoritmo Supervisionado
- É um dos algoritmos mais básicos e bem difundido do paradigma baseado em distância
- Em geral, faz o uso da distância euclidiana
- Para sua execução, definimos **a medida de distância** e o número de **k**

---

---

```
1 inicialização:
2     Preparar conjunto de dados de entrada e saída
3     Informar o valor de  $k$ ;
4 para cada nova amostra faça
5     Calcular distância para todas as amostras
6     Determinar o conjunto das  $k$ 's distâncias mais próximas
7     O rótulo com mais representantes no conjunto dos  $k$ 's
8     vizinhos será o escolhido
9 fim para
10 retornar: conjunto de rótulos de classificação
```

---



Classe 1

Classe 2

• ?

# Exemplo no Scikit-Learn

```
from sklearn.neighbors import KNeighborsClassifier # módulo neighbors classe KNeighborsClassifier
metric = 'euclidean'
k = 3
knn = KNeighborsClassifier(metric=metric, n_neighbors=k)

from sklearn.model_selection import train_test_split
X = data.drop('Target', axis=1)
y = data['Target']
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.3)

knn.fit(X_train, y_train) ## Entregamos os dados de treino
preds = knn.predict(X_val) ## Entregamos os dados sem rótulos e realizamos a classificação
print(confusion_matrix(y_val, preds))
```