Classificação

Identificando a qual categoria um objeto pertence

Classificação

Métodos

- Logistic Regression
- Stochastic Gradient Descent
- Naïve Bayes
- K-Nearest Neighbors
- Decision Tree
- Random Forest
- Support Vector Machine

Fonte: https://analyticsindiamag.com/7-types-classification-algorithms/

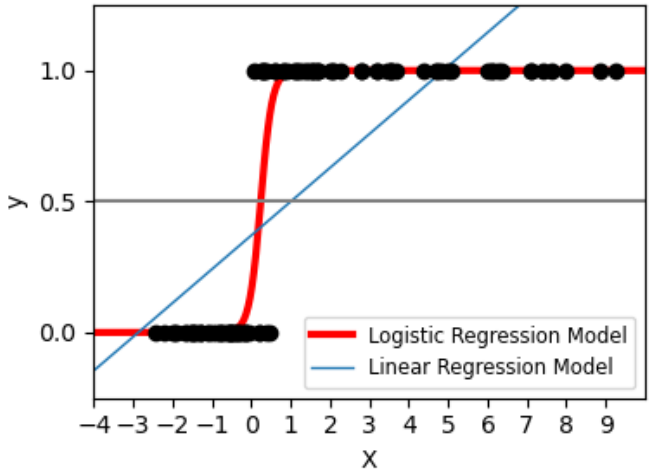
Regressão Logísitca

A regressão logística, apesar do nome, é um modelo linear para classificação, em vez de regressão. Uma curva em “S”

odds= p/ (1-p) = probability of event occurrence / probability of not event occurrence

ln(odds) = ln(p/(1-p))

logit(p) = ln(p/(1-p)) = b0+b1X1+b2X2+b3X3....+bkXk

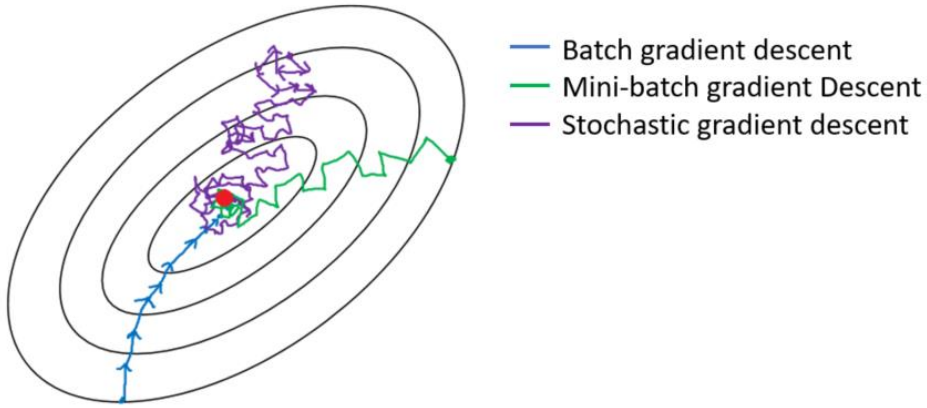


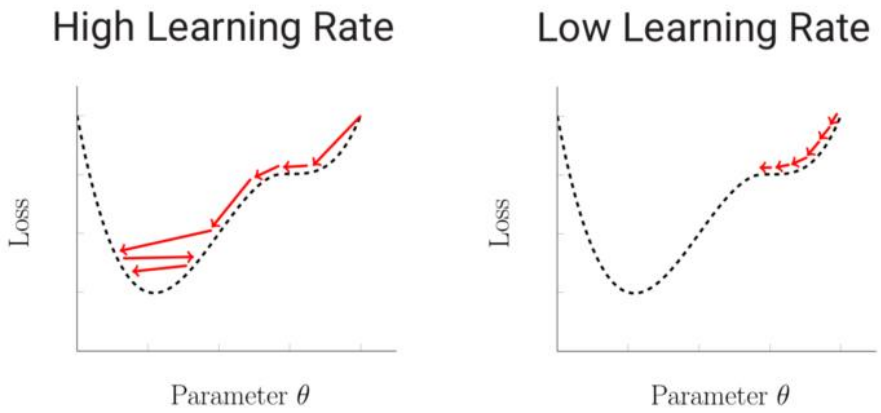
Fonte: https://scikit-learn.org/stable/auto_examples/linear_model/plot_logistic.html#sphx-glr-auto-examples-linear-model-plot-logistic-py

Stochastic Gradient Descent

A rigor, SGD é apenas uma técnica de otimização e não corresponde a uma família específica de modelos de aprendizado de máquina. É apenas uma forma de treinar um modelo.

É uma abordagem simples, mas muito eficiente para ajustar classificadores lineares e regressores em funções de perda convexa, como SVM (linear) e regressão logística





Fonte :https://scikit-learn.org/stable/modules/sgd.html
https://www.youtube.com/watch?v=vMh0zPT0tLI
https://medium.com/analytics-vidhya/gradient-descent-vs-stochastic-gd-vs-mini-batch-sgd-fbd3a2cb4ba4

Naive Bayes

Baseado no teorema de Bayes e assume a independência entre as features.

S: Spam
H: Ham (not spam)
B: 'Buy'
C: 'Cheap'

Naive Bayes

$$P(S \mid B \cap C) = \frac{P(B|S)\,P(C|S)\,P(S)}{P(B|S)\,P(C|S)\,P(S) + P(B|H)\,P(C|H)\,P(H)}$$

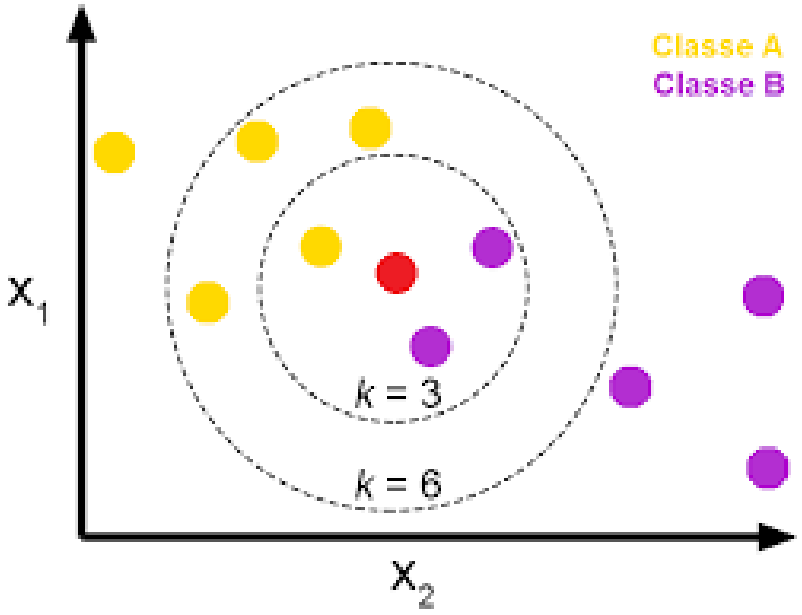
$$P(\text{spam if "Buy" \& "Cheap"}) = \frac{\frac{20}{25}\,\frac{15}{25}\,\frac{25}{100}}{\frac{20}{25}\,\frac{15}{25}\,\frac{25}{100} + \frac{5}{75}\,\frac{10}{75}\,\frac{75}{100}}$$

$$= 94.737\%$$

Fonte :https://scikit-learn.org/stable/modules/naive_bayes.html
https://www.youtube.com/watch?v=Q8l0Vip5YUw

K-Nearest Neighbors

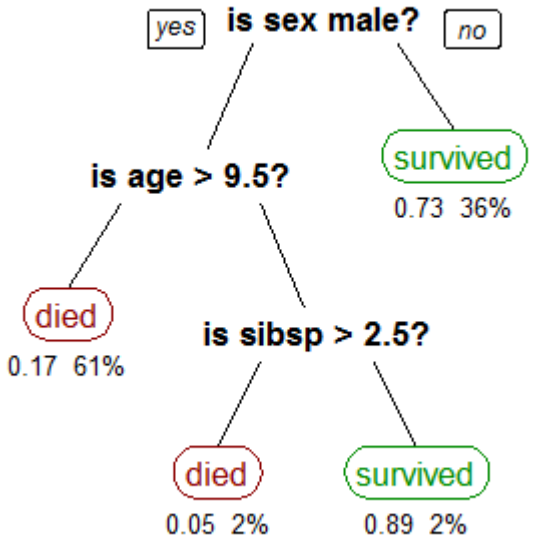
KNN funciona encontrando as distâncias entre uma amostra e os K exemplos mais próximos, em seguida, vota para o rótulo mais frequente (no caso de classificação) ou calcula a média dos rótulos (em o caso de regressão)



Fonte :https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761
https://medium.com/brasil-ai/knn-k-nearest-neighbors-1-e140c82e9c4e

Decision Tree

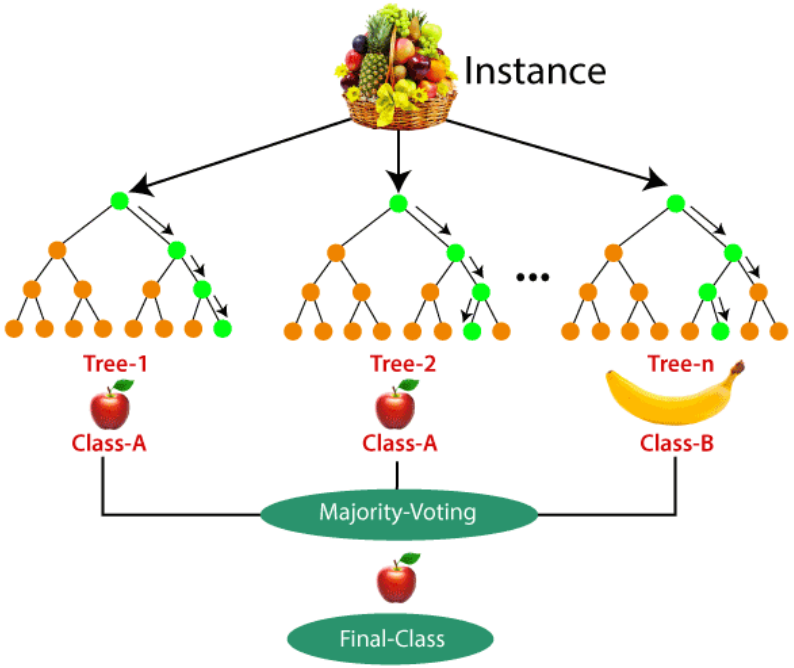
Todas os features são consideradas e diferentes pontos de divisão são experimentados e testados usando uma função de custo. A divisão com o melhor custo (ou menor custo) é selecionada.



Fonte :https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052
https://www.youtube.com/watch?v=7VeUPuFGJHk

Random Forest

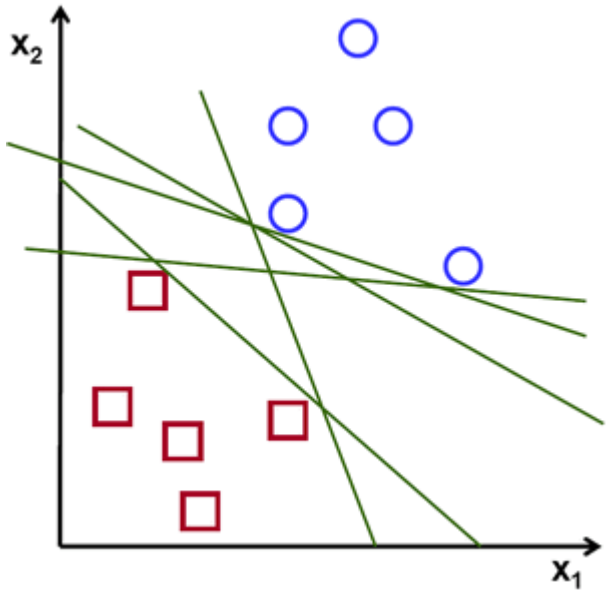
Random Forest é composta por várias Decision Trees. A união é feita por Bagging.

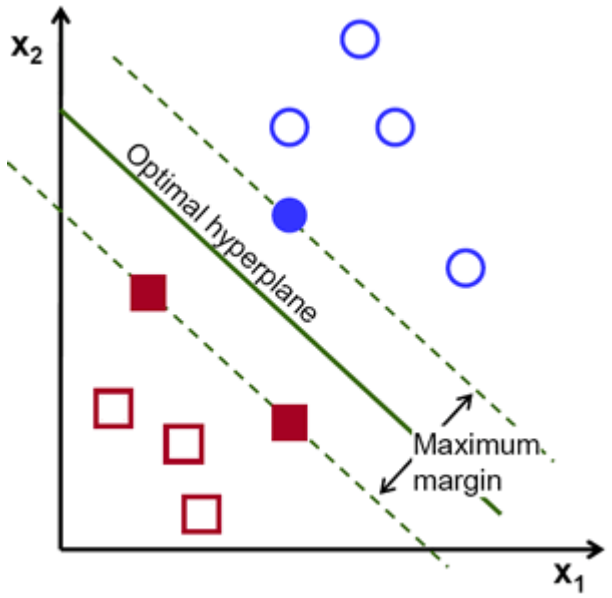


Fonte :https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/

Support Vector Machine

O SVM tenta aumentar a margem de distância entre amostra de diferentes grupos

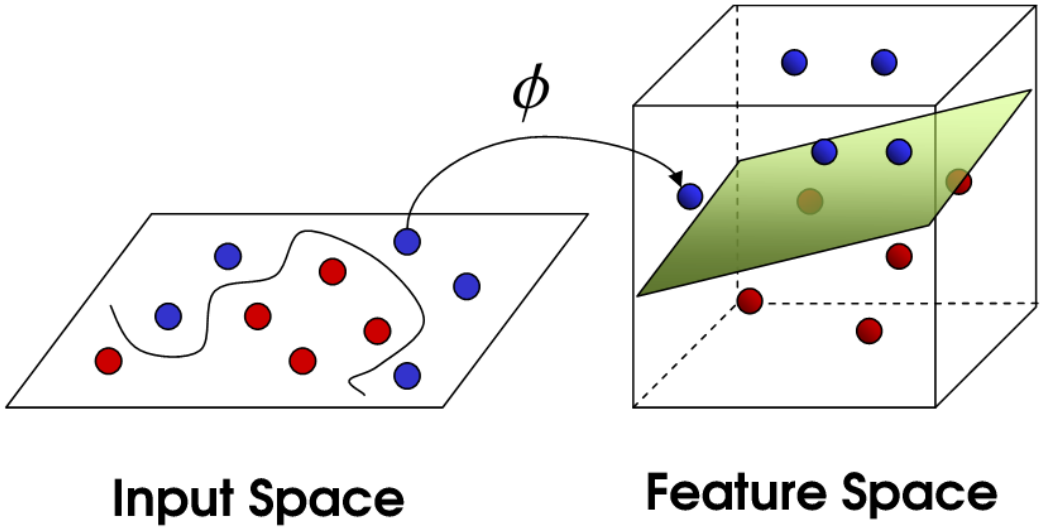




Fonte :https://scikit-learn.org/stable/modules/svm.html

Support Vector Machine

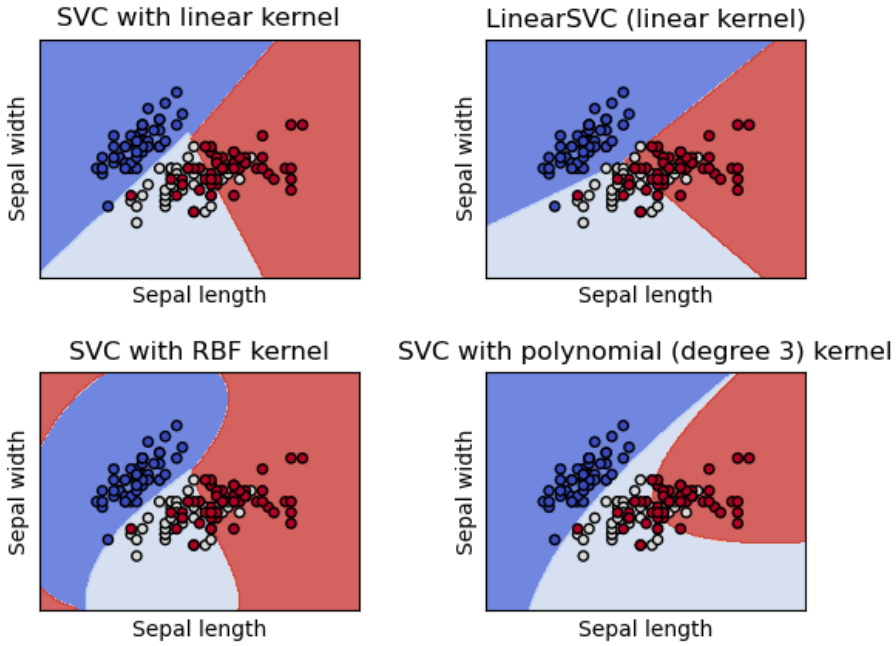
O truque do kernel é uma transformada que permite a aplicação de um plano em dados que originalmente não poderia ser separados desta forma



Fonte :https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

Support Vector Machine

O truque do kernel é uma transformada que permite a aplicação de um plano em dados que originalmente não poderia ser separados desta forma



Fonte:https://scikit-learn.org/stable/auto_examples/svm/plot_iris_svc.html