EDGARD JOSEPH KIRIYAMA

DATA ANALYTICS

MACHINE LEARNING COM PYTHON

AULA 05

SUMÁRIO

O QUE VEM POR AÍ?

Olá, jovem analytic!

Você está na terceira disciplina do curso e aqui começa mais um ponto importantíssimo da jornada em que está trilhando.

Na aula passada falamos sobre Feature Engineering e mostramos a importância de modelar os dados para conseguirmos atingir os resultados esperados, tanto na análise correta dos dados, bem como na construção de modelos de Machine Learning.

Chegamos em outro momento importante de nossa jornada! Vamos falar sobre modelos de Machine Learning com algoritmos de regressão.

Este é o momento de aprimorarmos e elevarmos o nosso nível! Vamos te mostrar o que são modelos de ML de regressão, quais são os modelos mais utilizados e como avaliar se o modelo está realizando as predições de forma correta.

Vamos entender como este aprendizado pode nos ajudar no nosso dia a dia! Bora lá?

CONHEÇA SOBRE O ASSUNTO

O Machine Learing (Aprendizado de Máquina), é um método de análise de dados que automatiza a construção de modelos analíticos.

Relembrando o que tivemos nas aulas passadas, os modelos de Machine Learning (ML) são classificados conforme ilustrado na figura 1 – Estrutura dos modelos clássicos de Machine Learning.

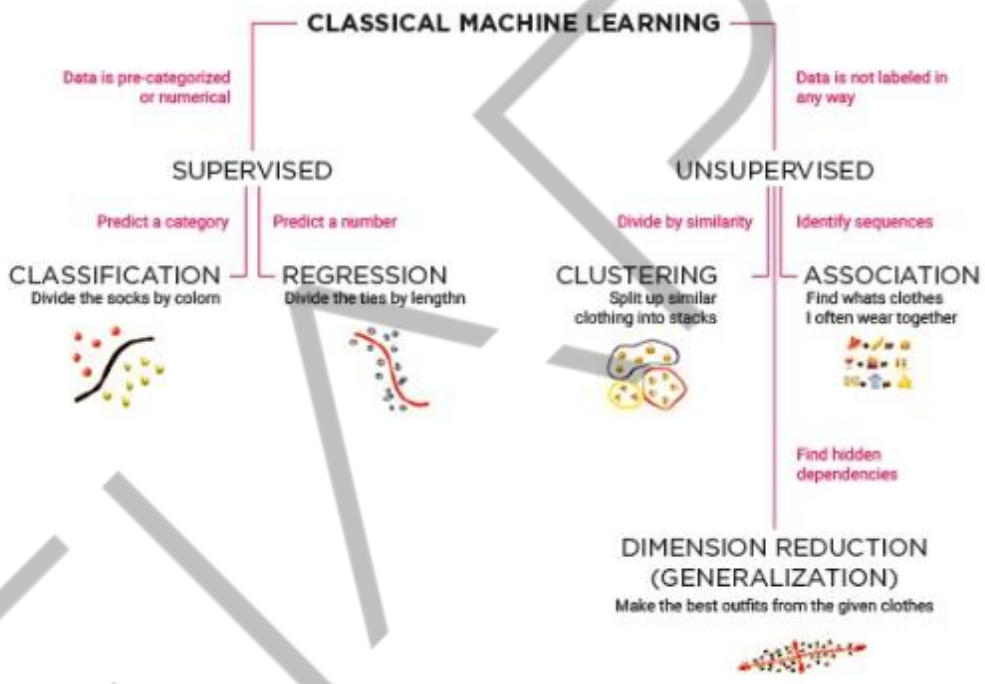

Figura 1 - Estrutura dos modelos clássicos de Machine Learning
Fonte: https://www.datageeks.com.br/machine-learning/ (2019), adaptado por FIAP (2023)

O algoritmo supervisionado elabora regras que podem fornecer a saída esperada para as entradas fornecidas. Essa capacidade de tomada de decisão permite que o algoritmo preveja novas entradas e tome decisões. Quando novos dados são encontrados, os dados e regras existentes são utilizados para entender e tomar decisões sobre eles. A fórmula é usada para prever variáveis dependentes (o que você deseja medir) a partir de variáveis independentes (do que você acha que sua medida de destino depende). Os valores previstos são contínuos. A regressão fornece resultados contínuos.

A regressão é um método de Machine Learning (ML) supervisionado. É um processo muito utilizado para realizar predição de variáveis numéricas contínuas.

Geralmente é utilizada na área de finanças e investimentos, para encontrar a relação entre uma variável dependente (target) e várias outras variáveis independentes.

Existem diversos algoritmos de regressão. Seguem alguns exemplos:

- (Mutiple) Linear Regression
- Support Vector Regression
- Regression Trees
- Random Forest
- KNN – K Nearest Neighbours

REGRESSÃO LINEAR – LINEAR REGRESSION

A regressão linear é um modelo estatístico que faz a relação linear entre duas (Regressão Linear Simples - RLS) ou mais (Regressão Linear Múltipla - RLM) variáveis, sendo uma variável dependente e outras variáveis independentes. Esta relação linear basicamente significa que, quando uma ou mais variáveis independente aumenta ou diminui, a variável dependente aumenta ou diminui.

O RLS é uma função de primeiro grau simples, logo, o objetivo do modelo de ML é entender o padrão do conjunto de dados e pode ser descrito por uma função de primeiro grau, com uma variável. Já o RLM terá mais de uma variável.

Vale ressaltar que há conceitos que precisam estar evidentes para nós:

- **Variável Preditora:** é a nossa variável independente, no qual influencia as demais variáveis que queremos encontrar.
- **Variável algo ou dependente (target):** é a variável que queremos prever. É o valor a ser buscado.

O cálculo dos coeficientes, em algoritmos de Regressão Linear, utiliza dois métodos: MMQ – Método dos Mínimos Quadrados e MQO – Métodos dos Quadrados Ordinários. Tais métodos vão buscar o melhor valor que os coeficientes possam atingir, através da diferença entre o valor predito pela função e o valor real.

Para avaliar se o modelo de regressão linear está realizando as predições corretamente e tendo uma boa performance, são realizados os seguintes testes:

- Realizar o teste F de Significância global;
- Realizar o teste de significância individuais ou p-values dos coeficientes;
- Avaliar o coeficiente R^2.

As bibliotecas mais famosas que auxiliam na construção do modelo de Regressão Linear são duas:

- Sklearn: LinearRegression.
- Statsmodels: statsmodels.apq as sm.

Suport Vector Regression

O Suport Vector Regression (SVR) é um modelo de aprendizado de máquina utilizado para a tarefa de regressão. O objetivo do SVR é encontrar uma função que seja capaz de prever valores numéricos de uma variável de saída com base em um conjunto de variáveis de entrada.

O SVR é uma extensão do Support Vector Machine (SVM) e utiliza os mesmos princípios do SVM para realizar a regressão. O modelo utiliza um conjunto de vetores de suporte para definir uma fronteira de decisão que separa as amostras em duas classes, sendo uma para valores positivos e outra para valores negativos. O objetivo do SVR é minimizar a distância entre a fronteira de decisão e os pontos de dados.

Para avaliar se o modelo de SRV está realizando as predições corretamente e tendo uma boa performance, são realizadas as seguintes avaliações:

- Métricas de desempenho: Erro quadrático médio (MSE), o erro absoluto médio (MAE) e o coeficiente de determinação (R^2);
- Se houve overfitting e underfitting;
- Ter atenção nos ajustes nos Hiperparâmetro;
- Comparação com outros modelos;
- Validação cruzada.

A biblioteca mais famosa que auxilia na construção do modelo de SVR:

- Sklearn: sklearn.svm (SVR)

Regression Trees

Regression trees são técnicas de aprendizado de máquina usados para realizar análises de regressão. Eles funcionam criando uma árvore de decisão que divide um conjunto de dados em subconjuntos menores, usando variáveis preditoras para prever uma variável de resposta contínua. A árvore é criada a partir de uma série de decisões binárias que separam os dados em grupos cada vez menores, até que o algoritmo encontre os melhores subconjuntos para cada grupo.

O objetivo é criar um modelo que preveja o valor de uma variável de destino aprendendo regras de decisão simples inferidas dos recursos de dados. Uma árvore pode ser vista como uma aproximação constante por partes.

Em cada nível de árvore (max_depth), serão geradas várias separações nos atributos avaliados. O critério para construção do modelo pode ser ajustado para determinar a métrica para a impureza que pode atrapalhar a interpretação do modelo e dificultando a compreensão do mesmo.

Para avaliar o modelo:

- Avaliar o coeficiente R^2;
- Realizar a validação cruzada;
- Avaliar a importância da variável como o Gini ou o ganho de informação, por exemplo;
- Executar e avaliar os gráficos de resíduos;
- Comparação com outros modelos;
- Overfitting.

A biblioteca mais famosa que auxilia na construção do modelo de Regression Trees:

- Sklearn: sklearn.tree (DecisionTreeRegressor)

KNN – K Nearest Neighbours

K-Nearest Neighbors (KNN) é um algoritmo de aprendizado de máquina supervisionado utilizado para classificação e regressão. A ideia por trás do KNN é simples: para classificar um novo ponto de dados, o algoritmo encontra os K pontos de dados mais próximos em um conjunto de treinamento e atribui a classe mais comum desses vizinhos ao novo ponto.

O parâmetro K, que representa o número de vizinhos a serem considerados, é um hiperparâmetro que deve ser ajustado. Quando K é muito baixo, o modelo pode ser muito sensível a ruído ou outliers nos dados de treinamento, enquanto que um K muito alto pode levar a um modelo super simplificado que ignora nuances importantes dos dados.

O KNN pode ser utilizado para classificar ou prever a saída numérica de um novo conjunto de dados, dependendo do tipo de problema. Embora seja um algoritmo simples, o KNN pode ser bastante poderoso em conjunto com a seleção de recursos e a normalização de dados.

Para avaliar o modelo, é necessário verificar:

- Escolha do valor de K para evitar Overfitting e Underfitting;
- Distância métrica;
- Tamanho do conjunto de treinamento;
- Balanceamento das classes;
- Validação cruzada k-fold ou a validação cruzada leave-one-out;
- Aplicação das métricas de desempenho: acurácia, precisão, recall, F1-score e curva ROC.

A biblioteca mais famosa que auxiliam na construção do modelo de KNN:

- Scikit-Learn: sklearn.neighbors (NearestNeighbors)

Random Forest

O Random Forest é um algoritmo de aprendizado de máquina que utiliza a técnica de ensemble learning para construir um modelo preditivo preciso e robusto.

Ele cria múltiplas árvores de decisão aleatórias, onde cada árvore é treinada em uma sub-amostra aleatória dos dados de treinamento e em um subconjunto aleatório das características. Em seguida, os resultados de todas as árvores são combinados para produzir uma predição final mais precisa e estável.

O algoritmo é uma técnica popular e poderosa para a classificação e regressão de problemas. Ele é amplamente utilizado em diversas áreas, como finanças, marketing, medicina e ciência de dados em geral. Além disso, ele apresenta vantagens como robustez contra dados ruidosos, facilidade de interpretação e escalabilidade para grandes conjuntos de dados.

Algumas das principais métricas a serem consideradas ao avaliar um modelo de Random Forest incluem:

- Acurácia;
- Recall;
- F1-score;
- Curva ROC;
- Matriz de Confusão;
- AUC (área sob a curva).

A biblioteca mais famosa que auxilia na construção do modelo de Random Forest:

- Sckiti-learn: sklearn.ensemble (RandomForestClassifier)

Após estudarmos como avaliarmos os nossos modelos, é muito importante ficarmos atentos a dois conceitos extremamente importantes: Underfitting e Overfitting.

Underfitting e overfitting são dois conceitos importantes na aprendizagem de máquina, que se referem a problemas de modelagem que podem ocorrer durante o treinamento de um modelo.

O underfitting ocorre quando o modelo não é complexo o suficiente para capturar as relações nos dados de treinamento. Isso resulta em um modelo que não é capaz de ajustar os dados de treinamento adequadamente e, portanto, também não é capaz de fazer previsões precisas em novos dados. Em outras palavras, o modelo

é muito simplista para representar adequadamente a complexidade dos dados de treinamento.

O overfitting ocorre quando o modelo é muito complexo e é ajustado muito bem aos dados de treinamento, capturando tanto o sinal quanto o ruído nos dados. Como resultado, o modelo pode ter um desempenho excelente nos dados de treinamento, mas geralmente não é capaz de fazer previsões precisas em novos dados. Em outras palavras, o modelo se ajusta demais aos dados de treinamento e não generaliza bem para novos dados.

O objetivo é encontrar um equilíbrio entre o underfitting e o overfitting, a fim de criar um modelo que capture adequadamente as relações nos dados de treinamento e possa generalizar bem para novos dados.

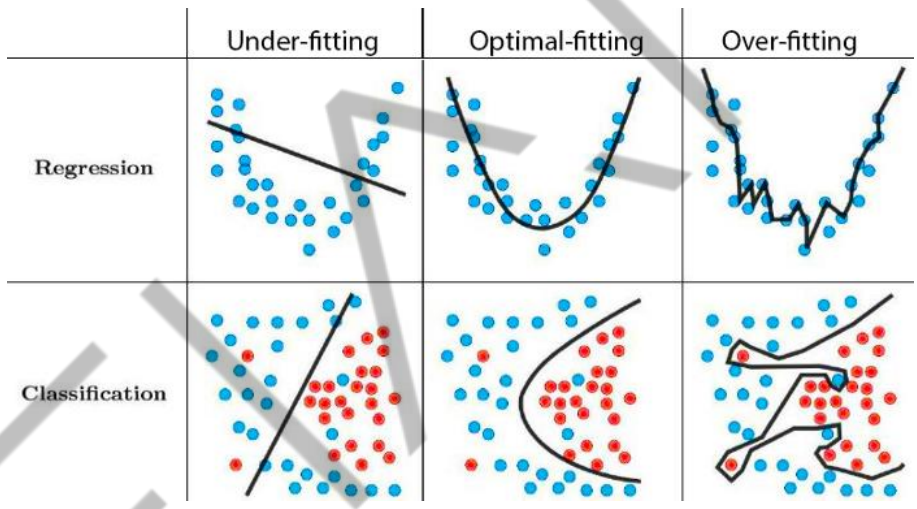

Figura 2 – Representação Gráfica de underfitting e overfitting.
Fonte: https://towardsdatascience.com/techniques-for-handling-underfitting-and-overfitting-in-machine-learning-348daa2380b9 (2021)

Que tal dar uma lida no artigo do Gabriel Stankevixno blog do medium? Ele traz uma boa visão e exemplos práticos de ML supervisionado de regressão:

Artigo sobre aprendizado supervisionado.

HANDS ON

Agora, chegou o momento de ver, na prática, como começar a importar os nossos dados e trabalhar com eles via programação. A ideia é não se limitar apenas ao código explícito no hands on, então é sempre bom procurar a documentação das bibliotecas, explorar novas funcionalidades e muito mais!

O QUE VOCÊ VIU NESTA AULA?

Introdução a Modelos de Regressão; Aplicação de Modelos de Machine Learning de Regressão; Tipo de modelos de regressão.

Daqui em diante, é importante que você replique os conhecimentos adquiridos para fortalecer mais suas bases e conhecimentos.

IMPORTANTE: não esqueça de praticar com o desafio da disciplina, para que assim você possa aprimorar os seus conhecimentos!

Você não está sozinho(a) nesta jornada! Te esperamos no Discord e nas lives com os nossos especialistas, onde você poderá tirar dúvidas, compartilhar conhecimentos e estabelecer conexões!

REFERÊNCIAS

DOCUMENTAÇÃO scikit-learng. **Disponível em: <https://scikit-learn.org/stable/modules/classes.html >.** Acesso em: 04 mai 2023.

DOCUMENTAÇÃO statsmodels. **Disponível em: <https://www.statsmodels.org/stable/api.html>.** Acesso em: 04 mai 2023.

GRUS, Joel. **Data Science do Zero**. Rio de Janeiro: Alta Books Editora, 2016.

HARRISON, Matt. **Machine Learning**: guia de referência rápida - trabalhando com dados estruturados em python. São Paulo: O'Reilly Media, 2019.

HASTIE, Trevor et al. **The Elements of Statistical Learning:** data mining, inference, and prediction. 2. ed. California: Springer, 2009.

PALAVRAS-CHAVE

Palavras-chave: Python. Machine Learning. Modelos de Regressão.