**Trabalho Final de Introdução ao Python**

## Kyojin

***Klysman Rezende, Felipe Jaworoski, Murilo Oliveira, Matheus Grieder.***

Abstract: Este trabalho tem como objeto de estudo a competição "Can You Predict Backorders?" disponibilizada no Kaggle. Dado o contexto do banco de dados e seu objetivo de previsão de estoque dos produtos, foi utilizado uma rede perceptron multicamadas (MLP), que é uma rede neural semelhante à perceptron (um tipo de rede neural artificial inventada em 1957 por Frank Rosenblatt), mas com mais de uma camada de neurônios em alimentação direta. Tal tipo de rede é composta por camadas de neurônios ligadas entre si por sinapses com pesos. O aprendizado nesse tipo de rede é geralmente feito através do algoritmo de retro-propagação do erro.

Palavras-chaves: gestão de inventário, MLP, InterpretML, blackbox, multilayers.

# 1 Introdução

Machine learning (ML) — ou aprendizado de máquina — consiste em um método que permite programas de computador tomarem decisões segundo a análise de dados baseada em determinadas regras e algoritmos. O aprendizado de máquina é um ramo essencial da inteligência artificial, é uma forma automatizada para algoritmos encontrarem informações interessantes em meio a uma grande quantidade de dados, sem a necessidade de especificar o que exatamente eles devem procurar. Ou seja, conforme são alimentados com mais dados, esses programas são capazes de aprender a reconhecer padrões e fornecer insights relevantes para as organizações. E é aí que está a importância do machine learning para as empresas. Neste estudo isso se faz intuitivo uma vez que o banco de dados é grande (1.35 milhões de linhas) e bem estruturado, contribuindo de maneira significativa para o aprendizado de máquina que foi utilizado. A medida que as ferramentas de ML coletam dados de dentro e de fora de uma organização, elas conseguem adquirir conhecimentos estratégicos para o negócio.

O principal componente do aprendizado de máquina é Dados: O aprendizado de máquina é basicamente sobre algoritmos e dados, mas os dados são considerados a chave para seu sucesso. É possível obter sucesso sem um bom algoritmo, mas se você não tiver dados suficientes, válidos e bem tratados, não obterá resultados relevantes.

O aprendizado de máquina pode ser dividido em duas grandes áreas:

* Aprendizagem supervisionada

Na aprendizagem supervisionada, o programa recebe um conjunto de dados para ser treinado e conseguir entender como eles se relacionam. Assim, quando o algoritmo precisar analisar novos dados, ele saberá o tipo de problema que precisará resolver e será capaz de chegar a conclusões por conta própria.

* Aprendizagem não supervisionada

Já na aprendizagem não supervisionada, o programa também recebe um conjunto de dados, mas não sabe exatamente o tipo de solução que deve procurar. Dessa forma, ele tenta descobrir algum relacionamento interessante entre os dados ou tendências que contribuam para facilitar a tomada de decisão.

## 1.1 Objetivo.

O problema de previsão de estoque é um tipo de aprendizagem supervisionada, o qual conhecemos, no treino, sua label - ou rótulo - isto é, cada exemplo em um conjunto de dados rotulado consiste em um ou mais recursos e um rótulo.

Para cada produto (sku - código identificador do material) do conjunto de teste, será necessário prever a probabilidade de falta no inventário. Disso, podemos concluir que o melhor modelo a ser implementado será aquele que se mostra competente em problemas do tipo classificação em aprendizado supervisionado

# 2 Materiais e métodos

## 2.1 Conjunto de dados

O conjunto de dados da competição contém dados históricos dos produtos ativos no inventário 8 semanas anteriores da semana que gostaríamos de prever, capturados como uma foto de todo o inventário no início da semana. Todas as amostras foram utilizadas no modelo, sendo isso uma condição crucial para eficiência do aprendizado.

Sabendo que o banco de dados foi modificado e processado para a competição atual, partiu-se direto para a técnica de engenharia de recursos (processo de determinar quais recursos/atributos podem ser úteis no treinamento de um modelo de machine learning).

O banco de dados foi dividido da seguinte maneira:

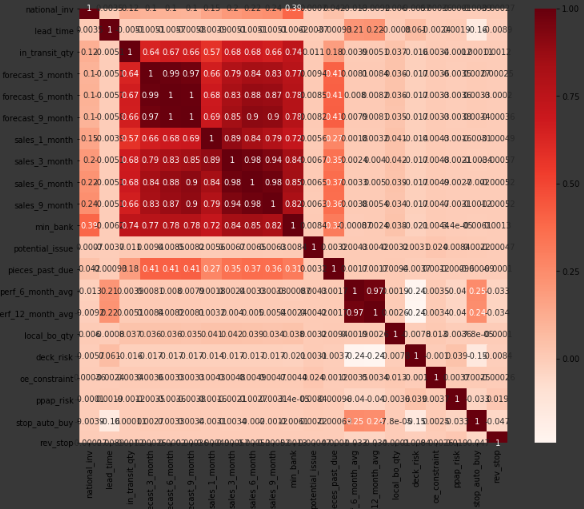


* sku - código identificador do material
* national\_inv - quantidade de material no inventário
* lead\_time - tempo de entrega registrado para o material
* in\_transit\_qty - quantidade em trânsito
* min\_bank - ponto de reposição
* pieces\_past\_due - peças a entregar em atraso
* local\_bo\_qty - quantidade requisitada não atendida
* forecast\_N\_month - previsão de vendas para os próximos N meses
* sales\_N\_month - quantidades vendidas nos últimos N meses
* perf\_N\_month\_avg - avaliação média do fornecedor do item nos últimos N meses
* Outros campos... - valores binários de classificação do material, relacionados ao risco definido no cadastro.

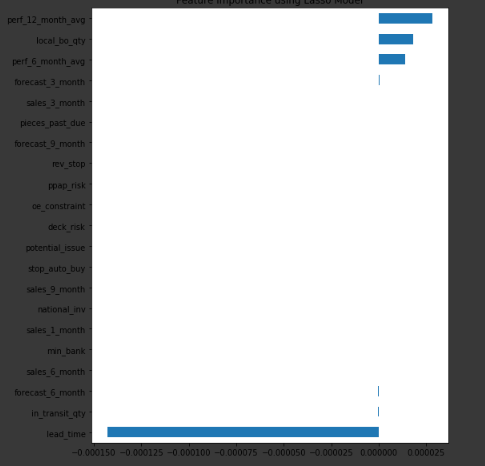
## 2.2 Engenharia de atributos

Dada o contexto citado no item anterior e utilizando bibliotecas como a Correlação de Pearson e LassoCV, foi possível concluir que as seguintes features são relevantes para um modelo de predição. São elas: per\_12\_month\_avg, local\_bo\_qty, lead\_time, e as forecasts. A abordagem InterpretML, que ainda será explicada, confirmou a relevância dessas features.

**Correlação de Pearson**



**LassoCV**



## 2.3 BaseLine

Baseline: Trata-se de um modelo usado como ponto de referência para comparar o desempenho de outro modelo (normalmente, mais complexo). Por exemplo, um modelo de regressão logística pode servir como uma boa linha de base para um modelo profundo.

Para um problema específico, o baseline ajuda os desenvolvedores do modelo a quantificar o desempenho mínimo esperado que um novo modelo deve atingir para que o novo modelo seja útil.

Para esse projeto foi escolhido como baseline o modelo de aprendizado Random Forest ou floresta aleatória, uma abordagem de conjunto para encontrar a árvore de decisão que melhor se ajusta aos dados de treinamento, criando muitas árvores de decisão e determinando a "média". A parte "aleatória" do termo refere-se à construção de cada uma das árvores de decisão a partir de uma seleção aleatória de recursos; a "floresta" refere-se ao conjunto de árvores de decisão.

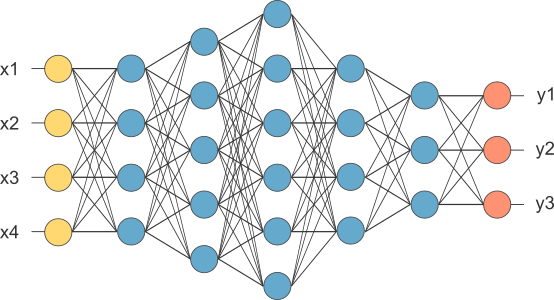
Neste caso a previsão foi de 0.91543 ROC como dempenho minimo: RandomForestClassifier(n\_estimators=30, random\_state=64, n\_jobs=-1). A curva ROC mostra o quão bom o modelo criado pode distinguir entre duas coisas (já que é utilizado para classificação). Essas duas coisas podem ser 0 ou 1, ou positivo e negativo. Os melhores modelos conseguem distinguir com precisão o binômio.

## 2.4 Seleção do modelo de aprendizado de máquina.

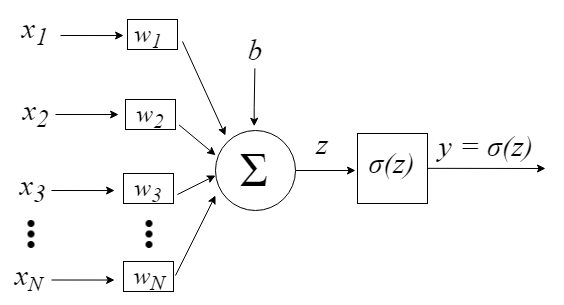
Rede Neural Perceptron Multicamadas (MLP)

Está é uma rede neural com uma ou mais camadas ocultas com um número indeterminado de neurônios. A camada oculta possui esse nome porque não é possível prever a saída desejada nas camadas intermediárias. Tal tipo de rede é composta por camadas de neurônios ligadas entre si por sinapses com pesos. O aprendizado nesse tipo de rede é geralmente feito através do algoritmo de retro-propagação do erro.

De maneira ilustrativa, na imagem abaixo é possível observar a arquitetura de uma MLP:



O processo de treinamento de redes MLP utiliza o algoritmo backpropagation conhecido também como regra delta generalizada. Esse é um algoritmo primário para realizar descida de gradiente em redes neurais. Os valores de saída de cada nó são calculados (e armazenados em cache) em uma passagem direta. Em seguida, a derivada parcial do erro em relação a cada parâmetro é calculada em uma passagem para trás pelo gráfico. As respostas produzidas pelas saídas são comparadas com as respectivas respostas rotuladas. Disso são gerados desvios (erros), para aplicar a segunda fase do método backpropagation que é a backward (propagação reversa). Nessa fase as alterações dos pesos e limiares de todos os neurônios da rede são executadas.



Para medir o sucesso da rede, foi utilizado como métrica de avaliação os índices ROC e o AUC.

A curva ROC mostra o quão bom o modelo criado pode distinguir entre duas coisas (já que é utilizado para classificação). Essas duas coisas podem ser 0 ou 1, ou positivo e negativo. Os melhores modelos conseguem distinguir com precisão o binômio.

AUC é que uma maneira de resumir a curva ROC em um único valor, agregando todos os limiares da ROC, calculando a área sob a curva.

## 2.5 Parâmetros usados para otimizar a rede neural

MLPClassifier(hidden\_layer\_sizes=(40,40,20,40,30,40,50), activation='relu', alpha=0.0001, learning\_rate='adaptive', solver='adam', random\_state=42, max\_iter=200, shuffle=True, verbose=True)

Os números (40,40,20,40,30,40,50) refere-se a quantidade de camadas e neurônios presente em cada uma. Para esse caso temos 7 camadas escondidas.

activation='relu' - Função de ativação para a camada oculta.

alpha - Parâmetro de penalidade L2 (termo de regularização).

learning\_rate - 'adaptável' mantém a taxa de aprendizado constante como 'learning\_rate\_init', desde que a perda de treinamento continue diminuindo. Cada vez que duas épocas consecutivas falham em diminuir a perda de treinamento em pelo menos tol, ou falham em aumentar a pontuação de validação em pelo menos tol se a 'interrupção precoce' estiver ativada, a taxa de aprendizado atual será dividida por 5.

solver - 'adam' refere-se a um otimizador estocástico

random\_state - é a semente usada pelo gerador de números aleatórios para os pesos.

max\_iter - número máximo de iterações. O solucionador itera até a convergência (determinada por 'tol') ou esse número de iterações.

shuffle - se as amostras devem ser embaralhadas em cada iteração.

verbose - imprimir mensagens de progresso no console.

## 2.6 Hardware and software

OS:

Windows 10

Software:

Jupyter notebook - Ambiente computacional web, interativo para criação de documentos “Jupyter Notebooks”, um tipo de documento JSON (formato compacto, de padrão aberto independente, de troca de dados simples e rápida entre sistemas) com um esquema e contém uma lista ordenada de células que podem conter código, texto, fórmulas matemáticas, plotagens e imagens. A extensão dos notebooks é “.ipynb”.

Hardware:

Desktop com 64GB DDR4

Duas GPUs com 8 GB DDR6

CPU Intel Core i7-8850H

Número de núcleos 6

Litografia 14 nm

Nº de threads 12

Frequência baseada em processador 2.60 GHz

Frequência turbo max 4.30 GHz

Cache 9 MB Intel® Smart Cache

Velocidade do barramento 8 GT/s

# 3 Resultados e discussão

Um dos maiores problemas em aprendizado de máquina é dificuldade em explicar como funcionam os modelos e o que leva eles a tomarem certas decisões. Mesmo sabendo que os resultados são ótimos. Saber explicar seu comportamento, ou seja, por que está chegando a tais resultados é tão importante quanto acertar uma predição corretamente.

Intrepret ML - Microsoft

O InterpretML é um pacote python de código aberto para treinar modelos de aprendizado de máquina interpretáveis ​​e explicar os sistemas de caixa preta. A interpretabilidade é essencial para:

Depuração de modelo - Por que meu modelo cometeu esse erro?

Detectando viés - Meu modelo discrimina?

Cooperação humano-AI - Como posso entender e confiar nas decisões do modelo?

Conformidade regulamentar - Meu modelo atende aos requisitos legais?

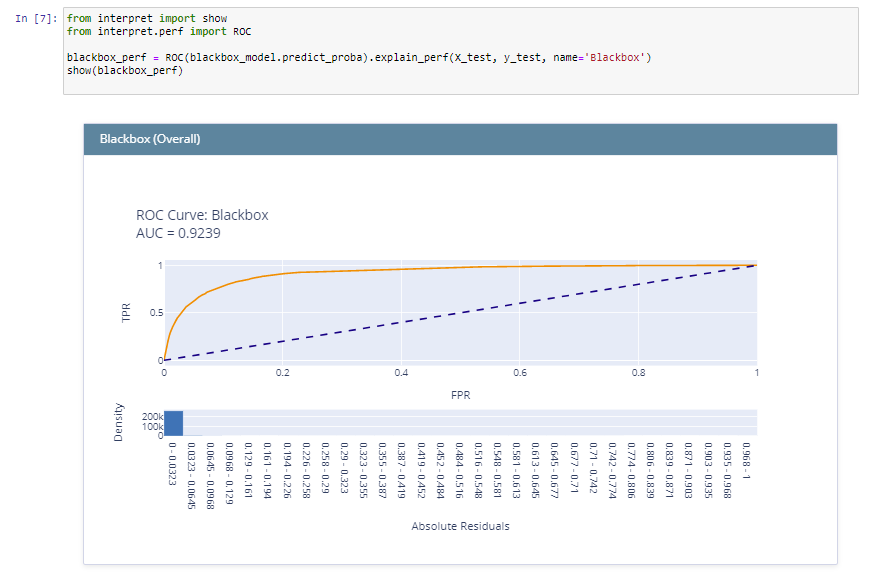
Aplicações de alto risco - Assistência médica, finanças, judiciário.

A Microsoft Research desenvolveu um algoritmo chamado EBM (Explicable Boosting Machine) \*, que possui alta precisão e capacidade de interpretação. A EBM usa técnicas modernas de aprendizado de máquina, como ensacamento e reforço, para dar nova vida aos GAMs tradicionais (modelos aditivos generalizados).

## 3.1 Mostra o desempenho do modelo de caixa preta - MLPClassifier

Nesse código podemos visualizar o desempenho da rede em função da curva ROC e AUC.

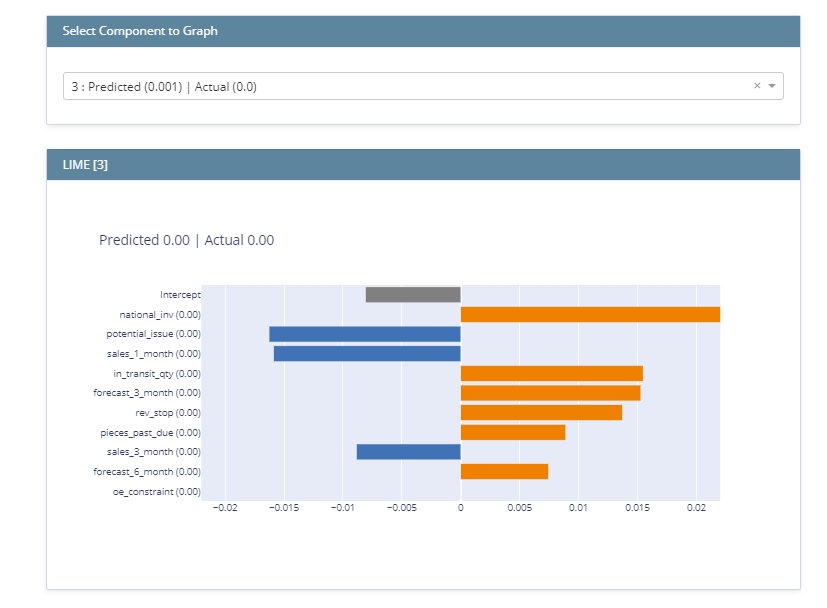
A AUC = 0.9239 não foi o maior valor atingido pela rede. Houve uma redução no número de épocas de 200 para 50 a fim de reduzir o tempo de processamento.



## 3.2 Explicações locais: Como uma previsão individual foi feita - MLPClassifier

A partir desse gráfico podemos observar como o modelo chegou em tal conclusão (predicted).

Isto é, do conjunto de teste, o elemento 3 tinha como valor real 0.0 e o modelo previu, com base nas features apresentadas no gráfico de barra horizontal, 0,001.

"Por que devo confiar em você ?: Explicando as previsões de qualquer classificador" (MT Ribeiro, S. Singh e C. Guestrin 2016)

## 3.3 Explicações globais: Como o modelo se comporta de maneira geral **- MLPClassifier**

