U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Inteligência artificial

Supervised learning

Caio Nogueira - up201806218
Carlos Lousada - up201806302
Miguel Silva - up201806388

Supervised learning

Aprendizagem supervisionada (*supervised learning*) consiste num tipo de *machine learning,* que pretende encontrar um modo de mapear *inputs* para *outputs*, partindo de um conjunto de dados rotulado. A previsão é feita a partir da classificação dos dados em determinadas *labels*.

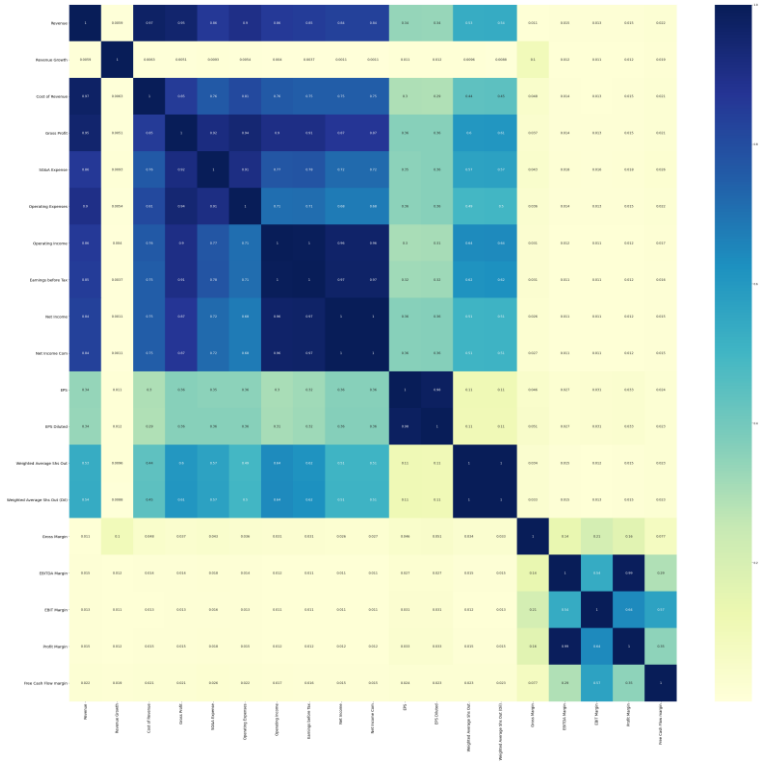
No contexto deste trabalho prático, o objetivo é, utilizando este mecanismo, prever a subida ou descida do preço das ações das empresas (no setor de tecnologia) do mercado de ações norte americano no ano de 2019. Este processo utiliza como dados o formulário 10-k relativo ao ano de 2018. Este relatório anual contém informações abrangentes acerca do desempenho financeiro das respetivas empresas.

Bibliotecas utilizadas: pandas (extração e manipulação dos dados), matplotlib (gráficos) , numpy (álgebra linear), sklearn (algoritmos de aprendizagem)

Dataset: https://www.kaggle.com/cnic92/beat-us-stock-market-data

Pré-processamento dos dados

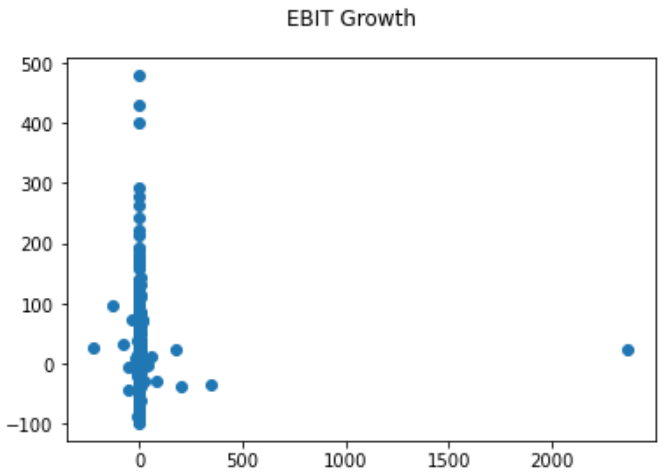
- O conjunto de dados usado para este problema de classificação é relativamente curto, contendo informações acerca de 638 empresas, sendo todas elas do setor de tecnologia. A usabilidade do *dataset* é muito alta, sendo que não existem valores nulos ou NaN. Contudo, foi necessário fazer merge dos dois ficheiros providenciados, dado que os dados relativos aos preços de 2019 continham informações acerca de empresas inexistentes no ficheiro de 2018.
- Dado que o conjunto de dados continha diversas colunas, foi feita uma análise relativamente à matriz de correlação dos dados, de modo a remover informação repetida e desnecessária ao classificador.

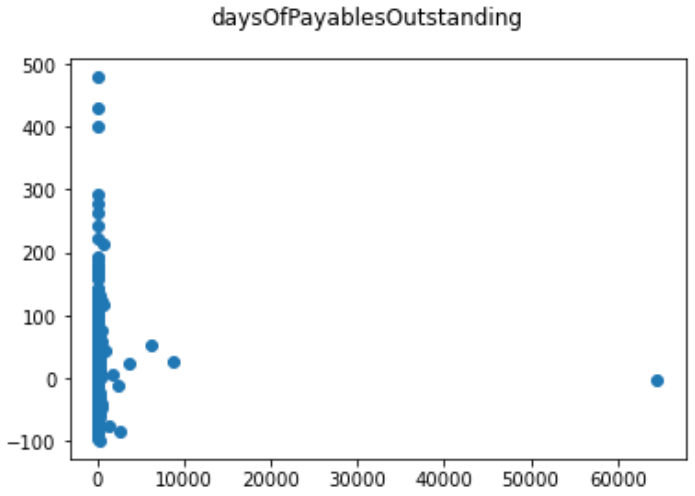


Matriz de correlação
(apenas contém as primeiras 20 colunas)

Pré-processamento dos dados

- Para além da análise através da matriz de correlação, foi também feita a remoção dos ***outliers*** existentes do conjunto de dados. De modo a auxiliar a sua deteção, foram feitos *scatter plots* de cada um dos atributos relativamente à variação do preço das ações em 2019. Os valores que se distinguiam em demasia dos restantes foram substituídos pela média dos restantes valores para esse atributo. As figuras seguintes contêm exemplos de *outliers* tratados.





Modelos de aprendizagem

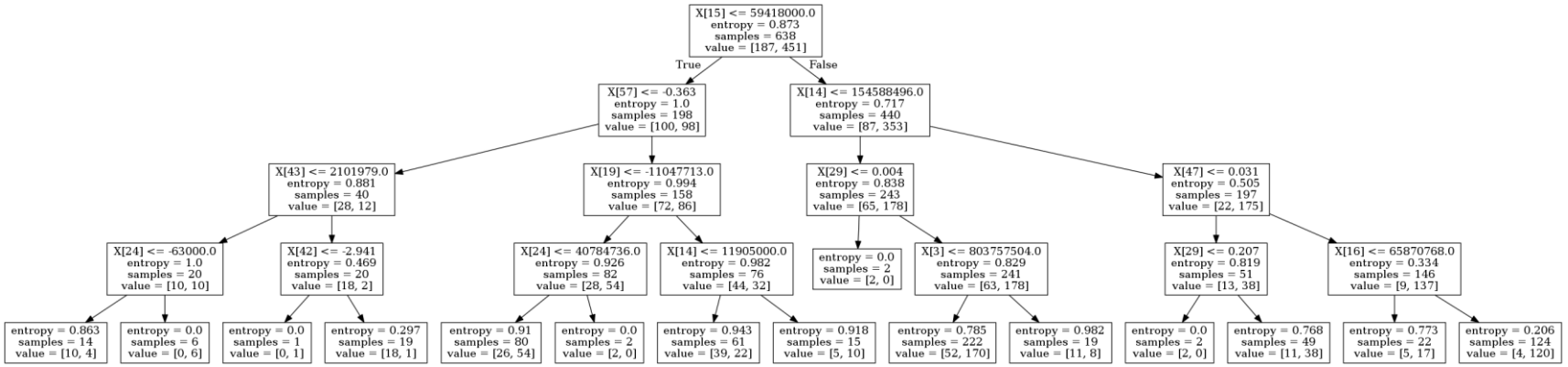
- Os modelos que escolhemos desenvolver no âmbito deste projeto são: as árvores de decisão, redes neuronais, *support vector machines e k-nearest neighbours.* Os modelos implementados usam *cross-validation* para treino, através da classe *StratifiedKFold* do módulo *sklearn.model_selection*. A escolha dos hiperparâmetros é feita através de *GridSearchCV.*
- **Redes neuronais:**
  - O modelo de redes neuronais usado será uma *multi-layer neural network,* usando o classificador implementado pela classe MLPClassifier do módulo sklearn.neural_network. Os parâmetros que influenciam o resultado da classificação são o número de *hidden layers,* função de ativação e curva de aprendizagem *(learning rate).*

Modelos de aprendizagem

- **Support vector machines (SVM):**
  - SVM é um classificador que pretende encontrar uma linha de separação (hiper plano) entre as instâncias das diferentes classes, de modo a separar os seus dados, tentando maximizar a separação entre elas. De modo a aumentar a eficiência do classificador, serão usados diferentes tipos de kernel's.
  - Será usado o classificador *svm* do módulo *sklearn.*
- **K-nearest neighbours (KNN):**
  - O algoritmo KNN permite obter uma classificação simples que não necessita de treino para a geração do modelo, dado que usa apenas as instâncias dos vizinhos mais próximos, de modo a classificar um certo conjunto de dados.
  - Será usado o classificador *KNeighborsClassifier* do módulo *sklearn.neighbours.*

Modelos de aprendizagem

- **Árvores de decisão (já implementado)**
  - Modelo de classificação que consiste em dividir um conjunto de dados em subconjuntos sucessivamente mais pequenos, através de uma estrutura em árvore.
  - De modo a aumentar a precisão do classificador, foi usada uma *GridSearchCV* (com 10 splits), de modo a encontrar os melhores parâmetros para um determinado conjunto de dados. Os hiperparâmetros analisados são o número de *features,* profundidade, critério de divisão. A precisão atingida através deste classificador é aproximadamente 72%.



Referências

- Slides do Moodle
- https://www.saedsayad.com/decision_tree.htm
- https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python
- https://www.investopedia.com/terms/1/10-k.asp