Trabalho prático final
Desenvolvimento e interpretação de modelos preditivos com AM

Contexto e objetivo do trabalho

O trabalho prático da disciplina **CMP263 - Aprendizagem de Máquina** visa permitir que os alunos desenvolvam um modelo preditivo para um problema de interesse, praticando aspectos discutidos na disciplina relacionados ao treinamento e avaliação de modelos de classificação ou regressão, e interpretação dos modelos gerados.

A proposta do projeto final é que os alunos aprofundem e consolidem sua experiência no desenvolvimento de modelos preditivos, abordando aspectos ao longo de toda a metodologia de treinamento de modelos, conforme discutimos em aula. Esta metodologia envolve:

i) *análise exploratória dos dados*, para identificar possíveis problemas nos dados que possam impactar negativamente no treinamento de modelos;

ii) *pré-processamento dos dados*, abordando aspectos como correção de outliers e de valores faltantes, codificação de atributos categóricos, discretização de atributos numéricos, normalização, ajuste de desbalanceamento de classes e redução de dimensionalidade

iii) *treinamento e validação dos modelos*, utilizando as melhores práticas em relação a estratégias de divisão de dados para otimização de hiperparâmetros e seleção de modelos

iv) *interpretação do modelo treinado*, buscando obter insights sobre o impacto dos atributos na tomada de decisão

Os alunos deverão selecionar um conjunto de dados e uma pergunta de pesquisa de interesse que envolva uma tarefa de classificação ou regressão, e então propor e implementar uma solução atendendo ao máximo possível às boas práticas na metodologia de AM supervisionado, conforme discussões em aula. As seções a seguir detalham alguns aspectos relacionados ao desenvolvimento do trabalho.

O trabalho deverá ser realizado **individualmente**.

Desenvolvimento: Dados e pré-processamento

Pode ser utilizado qualquer conjunto de dados de interesse dos alunos, incluindo dados provenientes de seus projetos de mestrado ou doutorado.

Dado que o interesse do projeto é explorar a metodologia de AM supervisionado, incluindo a etapa de pré-processamento dos dados, é interessante que os dados a serem utilizados não sejam "perfeitos" (isto é, que não se encontrem já completamente pré-processados e prontos para a modelagem preditiva). É interessante que os alunos tenham a oportunidade de empregar 2-3 estratégias de pré-processamento de dados dentre as que foram discutidas em aulas, como por exemplo: detecção e correção de outliers, correção de valores faltantes, normalização, codificação e discretização de atributos, ajuste de desbalanceamento de classes, e redução de dimensionalidade.

A tarefa pode ser de classificação ou regressão.

Recomenda-se que os alunos implementem a estratégia de *holdout* para gerar um conjunto de treinamento que será utilizado para o desenvolvimento do modelo (por exemplo, otimização de hiperparâmetros) e um conjunto de teste a ser utilizado na análise de desempenho do modelo final.

Também recomenda-se o uso do recurso de **Pipeline**, do Python, o qual pode ser facilmente integrado ao treinamento e avaliação de modelos com validação cruzada, evitando problemas de *data leakage*. Todas as referências utilizadas na coleta de dados e para desenvolvimento das estratégias de análise de dados devem ser citadas no relatório.

Como sugestão para os alunos, alguns repositórios que podem ser buscados:

Kaggle: https://www.kaggle.com/

UCI ML Repository: https://archive.ics.uci.edu/ml/datasets.php

OpenML: https://www.openml.org/

Desenvolvimento: Algoritmos e Avaliação dos modelos

Os algoritmos de aprendizado supervisionado a serem aplicados também são de livre escolha dos alunos, mas recomenda-se que a escolha englobe algoritmos que foram discutidos no curso. Sugere-se que os alunos selecionem 3 a 4 algoritmos para realizar a otimização de hiperparâmetros, assim treinando melhores modelos preditivos para os dados de interesse..

Para otimização de parâmetros e avaliação de modelo, os alunos deverão explorar estratégias de divisão de dados discutidas em aula. Será levada em consideração na avaliação do projeto a escolha da estratégia da divisão de dados. Recomenda-se o uso de validação cruzada k-fold (ou validação cruzada aninhada, quando possível) para uma avaliação mais robusta dos modelos e menos suscetível a *data leakage*.

As métricas de desempenho a serem empregadas são de livre escolha dos alunos. Sugere-se que seja feita a i) escolha de uma métrica alvo (aquela a ser maximizada ou minimizada), para guiar a seleção de modelos e ii) quando pertinente, se reporte no relatório o desempenho dos modelos de forma mais completa, incluindo outras métricas interessantes de analisar no domínio do problema escolhido e a matriz de confusão (para tarefas de classificação). A escolha da métrica alvo deve ser coerente com o problema de pesquisa, e também será avaliada.

Desenvolvimento: Sumarização dos resultados

O objetivo da sumarização dos resultados é identificar a distribuição de desempenho de um determinado algoritmo para a tarefa de predição selecionada. Para os processos iterativos implementados (por exemplo, validação cruzada), sugere-se o uso de sumarização por média e desvio padrão do desempenho, bem como gráficos do tipo box plot, violin plot, joy plot, etc, para visualização da distribuição de desempenho por modelo ao longo de *n* execuções. O uso de gráficos é importante pois viabiliza uma análise visual dos resultados.

Para o desempenho do modelo final com os dados de teste, os alunos podem reportar a(s) métrica(s) de desempenho resultante(s) e, quando pertinente, mostrar a matriz de confusão.

Desenvolvimento: Interpretação dos modelos

Os alunos deverão explorar algum método de interpretação de modelos e/ou modelos naturalmente interpretáveis (por exemplo, árvores de decisão), a fim de compreender ou extrair hipóteses sobre quais atributos são aparentemente mais relevantes para a tarefa de predição e/ou como eles impactam na decisão do modelo. Sugere-se que se faça esta investigação apenas para um modelo, a ser escolhido com base no seu desempenho preditivo (isto é, o melhor modelo conforme avaliação dos alunos). Os alunos deverão incluir no relatório informações obtidas desta análise, como gráficos, tabelas, etc, e discutir a respeito das relações encontradas na análise que mais chamaram a atenção, seja pela pertinência da associação ou por ser um resultado inesperado.

Desenvolvimento: ambiente de trabalho

Recomenda-se que o trabalho seja desenvolvido em Python ou R, em razão da ampla disponibilidade de bibliotecas ou pacotes que implementam as funcionalidades necessárias para pré-processar dados e desenvolver modelos de

AM. Para Python, o uso do Google Colab é desejável visto que facilita a reprodutibilidade do trabalho.

Relatório final

O relatório deve ser entregue em PDF, contendo um breve título para o trabalho, identificação do aluno, e uma descrição clara e objetiva dos seguintes itens:

- Dados utilizados e objetivo da análise, isto é, qual tarefa o modelo preditivo deveria ser capaz de resolver. Detalhes sobre os dados utilizados, como origem, tipo de tarefa, dimensionalidade, tipos dos atributos, problemas identificados nos dados, etc.
- Resumo das etapas de pré-processamento aplicadas. Não precisam ser explicados os métodos utilizados, apenas citadas as escolhas feitas nesta etapa de preparação dos dados.
- Resumo dos algoritmos utilizados e estratégias adotadas para otimização de hiperparâmetros e avaliação/seleção de modelos (divisão de dados e métricas). No caso de otimização de hiperparâmetros, é interessante citar os valores testados.
- Resultados da análise de desempenho obtida na etapa de otimização de hiperparâmetros, e para o modelo final (utilizando os dados de teste)
- Resultados sobre a interpretação do modelo final
- Conclusões do trabalho em relação aos resultados obtidos e ao sucesso ou às dificuldades em abordar a tarefa de predição escolhida
- Link para o Google Colab (desejável)
- Fontes consultadas na execução do trabalho

Entregáveis e Prazo

- Relatório final
- Código fonte, sendo fortemente indicado o uso de Google Colab. Neste caso, pode ser disponibilizado apenas o link de acesso.
- Apresentação oral

A entrega de relatório e código deverá ser realizada pelo Moodle da disciplina, com prazo final em 16 de setembro de 2022, às 23:59h. O trabalho deverá ser apresentado oralmente em aula, nos dias 19/09 ou 21/09, em data específica a ser marcada pela professora. A apresentação deverá durar **no máximo 12 minutos**.

Dúvidas Em caso de dúvidas sobre a realização do trabalho, incluindo a definição do tema de pesquisa, entre em contato com a professora através do e-mail mrmendoza@inf.ufrgs.br ou pelo Moodle da disciplina.