

PLANO DE ENSINO

IDENTIFICAÇÃO		
HABILITAÇÃO:	Classificação, Clusterização e Regressão de Dados	C. H. TOTAL: 80 horas
PROFESSOR (A):	Gerson do Nascimento Silva (PhD Student, UnB)	

PLANEJAMENTO			
BASES TECNOLÓGICAS			
<p>Ementa: Desenvolvimento de código-fonte (<i>script</i>) utilizando linguagem <i>python</i> para implementação, mostrando como:</p>			
<p>1. Classificação, Clusterização e Regressão</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; vertical-align: top;"> <ul style="list-style-type: none"> a) Criar e otimizar modelo para regressão e classificação; b) Utilizar “<i>nearest neighbours</i>” para regressão e classificação; c) Fazer agrupamento aglomerativo (<i>Agglomerative Clustering</i>); d) Fazer clusterização com o “<i>Kmeans</i>”; e) Fazer clusterização baseado em afinidade; f) Utilizar “<i>DBSCAN Clustering</i>”; g) Utilizar a abordagem do deslocamento médio (<i>MinShift</i>); h) Utilizar a árvore de classificação e regressão; i) Utilizar “<i>AdaBoost</i>”; j) Utilizar “<i>RandomForest</i>”; k) Utilizar “<i>GradientBoosting</i>”; l) Utilizar classificador e regressor multicamadas – “<i>MLP - Multi Layer Perceptron</i>”; m) Utilizar classificador e regressor de reforço gradual “<i>XgBoost</i>”; n) Utilizar classificador e regressor “<i>CatBoost</i>”; </td> <td style="width: 50%; vertical-align: top;"> <ul style="list-style-type: none"> o) Utilizar classificador e regressor “<i>LightGBM</i>”; p) Utilizar classificador e regressor “<i>SVM</i>”; q) Classificar com modelos lineares – “<i>Multiclass Classification</i>”; r) Classificar com modelos lineares – “<i>Naive Bayes</i>”; s) Classificar com modelos lineares – “<i>Nearest Neighbors</i>”; t) Classificar com modelos lineares – “<i>LDA e QDA</i>”; u) Classificar com modelos lineares – “<i>Tree Model</i>”; v) Classificar com modelos lineares – “<i>Ensemble Bagging Model</i>”; w) Classificar com modelos lineares – “<i>Ensemble Boosting Model</i>”; x) Utilizar métrica de classificação e regressão; y) Comparar algoritmos de classificação; z) Implementar “<i>Ensemble Model</i>”; aa) Salvar modelos treinados; bb) Avaliar modelos com curvas de aprendizagem; cc) Paralelizar execução e validação cruzada no “<i>XGBoost</i>”; dd) Otimizar número de árvores no “<i>XGBoost</i>”. </td> </tr> </table>		<ul style="list-style-type: none"> a) Criar e otimizar modelo para regressão e classificação; b) Utilizar “<i>nearest neighbours</i>” para regressão e classificação; c) Fazer agrupamento aglomerativo (<i>Agglomerative Clustering</i>); d) Fazer clusterização com o “<i>Kmeans</i>”; e) Fazer clusterização baseado em afinidade; f) Utilizar “<i>DBSCAN Clustering</i>”; g) Utilizar a abordagem do deslocamento médio (<i>MinShift</i>); h) Utilizar a árvore de classificação e regressão; i) Utilizar “<i>AdaBoost</i>”; j) Utilizar “<i>RandomForest</i>”; k) Utilizar “<i>GradientBoosting</i>”; l) Utilizar classificador e regressor multicamadas – “<i>MLP - Multi Layer Perceptron</i>”; m) Utilizar classificador e regressor de reforço gradual “<i>XgBoost</i>”; n) Utilizar classificador e regressor “<i>CatBoost</i>”; 	<ul style="list-style-type: none"> o) Utilizar classificador e regressor “<i>LightGBM</i>”; p) Utilizar classificador e regressor “<i>SVM</i>”; q) Classificar com modelos lineares – “<i>Multiclass Classification</i>”; r) Classificar com modelos lineares – “<i>Naive Bayes</i>”; s) Classificar com modelos lineares – “<i>Nearest Neighbors</i>”; t) Classificar com modelos lineares – “<i>LDA e QDA</i>”; u) Classificar com modelos lineares – “<i>Tree Model</i>”; v) Classificar com modelos lineares – “<i>Ensemble Bagging Model</i>”; w) Classificar com modelos lineares – “<i>Ensemble Boosting Model</i>”; x) Utilizar métrica de classificação e regressão; y) Comparar algoritmos de classificação; z) Implementar “<i>Ensemble Model</i>”; aa) Salvar modelos treinados; bb) Avaliar modelos com curvas de aprendizagem; cc) Paralelizar execução e validação cruzada no “<i>XGBoost</i>”; dd) Otimizar número de árvores no “<i>XGBoost</i>”.
<ul style="list-style-type: none"> a) Criar e otimizar modelo para regressão e classificação; b) Utilizar “<i>nearest neighbours</i>” para regressão e classificação; c) Fazer agrupamento aglomerativo (<i>Agglomerative Clustering</i>); d) Fazer clusterização com o “<i>Kmeans</i>”; e) Fazer clusterização baseado em afinidade; f) Utilizar “<i>DBSCAN Clustering</i>”; g) Utilizar a abordagem do deslocamento médio (<i>MinShift</i>); h) Utilizar a árvore de classificação e regressão; i) Utilizar “<i>AdaBoost</i>”; j) Utilizar “<i>RandomForest</i>”; k) Utilizar “<i>GradientBoosting</i>”; l) Utilizar classificador e regressor multicamadas – “<i>MLP - Multi Layer Perceptron</i>”; m) Utilizar classificador e regressor de reforço gradual “<i>XgBoost</i>”; n) Utilizar classificador e regressor “<i>CatBoost</i>”; 	<ul style="list-style-type: none"> o) Utilizar classificador e regressor “<i>LightGBM</i>”; p) Utilizar classificador e regressor “<i>SVM</i>”; q) Classificar com modelos lineares – “<i>Multiclass Classification</i>”; r) Classificar com modelos lineares – “<i>Naive Bayes</i>”; s) Classificar com modelos lineares – “<i>Nearest Neighbors</i>”; t) Classificar com modelos lineares – “<i>LDA e QDA</i>”; u) Classificar com modelos lineares – “<i>Tree Model</i>”; v) Classificar com modelos lineares – “<i>Ensemble Bagging Model</i>”; w) Classificar com modelos lineares – “<i>Ensemble Boosting Model</i>”; x) Utilizar métrica de classificação e regressão; y) Comparar algoritmos de classificação; z) Implementar “<i>Ensemble Model</i>”; aa) Salvar modelos treinados; bb) Avaliar modelos com curvas de aprendizagem; cc) Paralelizar execução e validação cruzada no “<i>XGBoost</i>”; dd) Otimizar número de árvores no “<i>XGBoost</i>”. 		
TEORIA	PRÁTICA		
<p>-video aula: (playlist do PhD Francisco A. Rodrigues – USP)</p> <ul style="list-style-type: none"> • “Estatísticas”; • “Probabilidades”; • “Redes Complexas”; • “Processos Estocásticos”; • “Ciência de Dados”; <p>-video aula: (playlist do Dr. Alexandre L. M. Levada – USP)</p> <ul style="list-style-type: none"> • “Introdução a Teoria dos Grafos”; • “Reconhecimento de Padrões”; <p>-video aula: (playlist da Dra. Cibele Russo – USP)</p> <ul style="list-style-type: none"> • “Visualização e Exploração de Dados”; • “Análise Multivariada e Aprendizado Não-Supervisionado”. 	<p>- Classificação: baseia-se em prever a categoria de uma observação dada. Procura-se estimar um “classificador” que gere como saída a classificação qualitativa de um dado não observado com base em dados de entrada (que abrangem observações com classificações já definidas). <u>Exemplo:</u> um classificador que utilize dados não observados de um paciente e classifique-o como doente ou não-doente.</p> <p>- Regressão: de forma similar a classificação, utiliza dados de entrada (preditores) já observados para prever uma resposta. A grande diferença é que, neste caso, procura-se estimar um valor numérico e não uma classificação de uma observação. <u>Exemplo:</u> estimar um modelo que utilize a idade e os anos de escolaridade de um indivíduo não-observado anteriormente para tentar prever seu salário. Utiliza-se como base desse modelo: idades, anos de escolaridades e salários de diversos indivíduos já observados anteriormente.</p> <p>- Agrupamento: também conhecido como “<i>Clustering</i>”, tem como objetivo agrupar observações em grupos conhecidos como “<i>clusters</i>”. Essas observações apresentam similaridades dentro de seu <i>cluster</i> e diferenças em relação aos demais <i>clusters</i> formados. Diferente da Classificação, não é realizada a rotulação dos <i>clusters</i>, fazendo com que não exista uma clusterização errada</p>		

	ou certa. A clusterização utilizada resulta em diferentes tipos de <i>clusters</i> , e a escolha dessas técnicas deve ser previamente analisada pelo pesquisador. <u>Exemplo</u> : agrupar fotos de animais similares em <i>clusters</i> , sem ter o conhecimento prévio de qual animal está sendo apresentado.

ANÁLISE DA REALIDADE
Pré-Requisito
<ol style="list-style-type: none"> 1. Lógica de Programação; 2. Programação Orientada a Objetos; 3. Análise de Sistemas Orientada a Objeto; 4. Fundamentos e Modelagem de banco de Dados.
Necessidade da turma:
- Programação intermediário em <i>python</i> ou outra linguagem de programação que traga subsídios para entendimento dos conceitos computacionais.

PROJEÇÃO DE FINALIDADES
Objetivos Gerais (Competências)
Compreender os conceitos de IA e seus recursos e capacidade para implementar código-fonte reproduzível.
Objetivos Específicos (Habilidades)
-Capacidade para implementar código-fonte (uso de <i>python</i> ou outras tecnologias com mesma finalidade);
-Entender como aplicar frameworks python focados em – Classificação;
-Entender como aplicar frameworks python focados em – Clusterização;
-Entender como aplicar frameworks python focados em – Regressão;
-Ser capaz de implementar código-fonte reproduzível utilizando conceitos anteriores.

FORMAS DE MEDIAÇÃO
Procedimentos Metodológicos
-Elaborar modelos de implementação de código-fonte que valide as teorias, com uso de exemplos;
-Propor trabalhos práticos sobre o assunto;
-Corrigir os trabalhos práticos;
-Elaborar avaliação do conhecimento sobre o assunto ministrado;
-Uso de computadores para implementar exemplos;

AVALIAÇÃO*	Instrumento(s)	Data
<ul style="list-style-type: none"> • Avaliação única por meio do instrumento <i>Quiz</i>; • Criação de um projeto final “<i>End to End</i>” implementado em código-fonte reproduzível python. 		Ao fim do módulo estudado.

* No Exame Quiz não é permitido qualquer espécie de acréscimo de nota (trabalhos, participações e outros) estranhos à avaliação.

* Quiz é um jogo mental no qual os jogadores tentam responder corretamente a questões que lhes são colocadas. A palavra também é utilizada como sinônimo de avaliação de aquisição de conhecimentos ou capacidades em ambientes de aprendizagem.

BIBLIOGRAFIA					
BIBLIOGRAFIA BÁSICA (Títulos, periódicos, etc)					
Título/Periódico	Autor	Edição	Editores	Ano	Livro Texto
Álgebra Linear	José Luiz Boldrini [et al.]	3ª edição	Harbra Ltda	1980	-
Trilhas Python	Eduardo Pereira	1ª edição	Casa do Código	2018	-

Python para Desenvolvedores	Luiz Eduardo Borges	3ª edição	Novatec	2014	Sim
BIBLIOGRAFIA COMPLEMENTAR (Títulos, periódicos, etc)					
Título/Periódico	Autor	Edição	Editora	Ano	
Curso Intensivo de Python	Eric Matthes	1ª edição	Novatec	2017	
Data Science from Scratch	Joel Grus	1ª edição	O'Reilly	2015	
Introduction to Applied Linear Algebra	Stephen Boyd	1ª edição	Cambridge University Press	2018	
Outros:					