Introduction to machine learning for scientific computing

PHD Course

CHAPTER 1: Introduction to Python

1.1. Installation for various operating systems: MaC OS X 3.x, Linux and Windows 10
1.2. Python working space: Scripts and import of scientific libraries
1.3. Basic commands: loops, functions, recursion, ...
1.4. Boolean and numerical operators.
1.5. Lists
1.6. Tuples, sets y dictionaries
1.7. Character strings
1.8. Files: Text, CSV y HTML
1.9. Classes: Some object oriented programming (optional if time permits)

CHAPTER 2: Preliminaries: The landscape of Machine Learning

2.1. What is Machine learning and types of machine learning strategies
2.2. Supervised learning: Regression y classification
2.3. The process of machine learning:
    2.3.1.- Data collection and preparation
    2.3.2.- Feature selection
    2.3.3.- Algorithm choice
    2.3.4.- Parameter and model selection
    2.3.5.- Training
    2.3.6.- Evaluation
2.4. Main elements
    2.4.1.- Inputs
    2.4.2.- Weights
    2.4.3.- Outputs
    2.4.4.- Targets
    2.4.5.- Activation function
    2.4.6.- Error
2.5. La maldición de la dimensionalidad
2.6. Testing machine learning algorithms:
    2.6.1.- Overfitting
    2.6.2.- Training, Testing and Validation Sets
    2.6.3.- The confusion matrix
    2.6.4.- Accuracy metrics
    2.6.5.- Receiver Operator Characteristic curve (ROC)
2.7. Basic Statistics: Averages, Variance and covariance, and the Gaussian

CHAPTER 3: Linear Algebra and learning from data

3.1. The least-squares fitting
3.2. Normal equations: Gram Matrix
3.3. Example: The stiffness matrix
3.4. Regularization in l2 and l1 norms
3.5. Covariance matrix reprise
3.6. Example: White noise removal: Wiener filter, Rudin-Osher-Fatemi Total variation model.
3.7. Gaussian convolution
3.8. Compressive sensing: LASSO regression
3.9. Dimensionality reduction: SVD and Principal Component Analysis
3.10. Python implementation of academic examples

CHAPTER 4: Introduction to Neural Networks

4.1. Neuronal models: McCulloch and Pitts
4.2. Neural Networks: the Perceptron
    4.2.1.- The learning rate
    4.2.2.- The bias input
    4.2.3.- The perceptron algorithm
    4.2.4.- Example: The logic functions, Python implementation

4.3. The Multilayer Perceptron (MLP)
    4.3.0.- The learning function F: learn from examples
    4.3.1.- Continuous Piecewise Linear learning Functions (CPL)
    4.3.2.- The nonlinear step: The activation function, ReLU and logistic functions
    4.3.3.- Composition of CPL functions
    4.3.4.- The training set
    4.3.5.- The testing set
    4.3.6.- The validation set
    4.3.7.- Iterative algorithm and convergence: stochastic gradient descent (SGD)
    4.3.8.- Estimation of the error: backpropagation algorithm and chain rule
    4.3.9.- Convolutional neural networks based on Toeplitz matrices

CHAPTER 5: Implementing ML-algorithms with NumPy, Scikit-Learn y TensorFlow

5.1. Installation of NumPy and Scikit-Learn in Python.
5.2. Implementing linear, logistic and softmax regression
5.4. Installation of TensorFlow in Python
5.5. Computation Graphs
5.6. Training of a multilayer neural network (deep network) with TensorFlow

Notes and remarks:

1.- We will teach this course in English, but we will use in class extra comments in Spanish to answer and clarify questions formulated by locals.

2.- The content of this course is very wide to cover in depth all mentioned themes. Our purpose is to address specific topics that might be the interest of particular students. Each student may choose the appropriate path of the course content to benefit the prospective use in her/his research area.

3.- We intentionally avoid the statistical-learning approach that we will mention tangentially.