THE UNIVERSITY OF MELBOURNE

Workshop 2

COMP90051 Machine Learning

Semester 2, 2018

Learning Outcomes

At the end of this workshop you should:

1. Be familiar with the Python ecosystem for ML
2. Be able to implement linear regression from scratch using NumPy (Worksheet 2a)
3. Develop intuition about the role of the prior and posterior in Bayesian inference (Worksheet 2b)

The SciPy stack

Scientific Python stack

NumPy

- Library for working with large multidimensional arrays
- High-level functions for arrays

native Python slow because interpretative.

SciPy

builds on NumPy

- Scientific computing library
- Functionality includes: statistics/random number generation, linear algebra, optimisation, special functions

scikit learn

- Machine learning library
- Includes implementations of most models covered in this course (exception: neural nets)

interfaces closely w/ Numpy

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$

- Library for analysis and manipulation of tabular data
- Provides similar functionality to DataFrames and dplyr in R

– Data cleaning

matplotlib

- 2D plotting library
- Provides similar interface to MATLAB

Other libraries in the ecosystem

written in Python or have APIs.

TensorFlow

allows you to work w/ tensors

- Library for numerical computations using data flow graphs
- Often used for neural nets
- Supports GPU, TPU acceleration

low level

PyStan

- A probabilistic programming language written in C++
- Great for smaller-scale statistical modelling
- Has a Python API

APACHE Spark™

- Cluster computing framework
- Supports scalable machine learning through Spark MLlib
- Has a Python API
- ML for big data

K Keras

- High-level neural net library written in Python
- Supports various backends: TensorFlow, CNTK and Theano

PyMC3

- A probabilistic programming language written in Python and built on top of Theano
- More “Pythonic” than Stan

Worksheets

- Download worksheets 2a & 2b from the LMS
- Work at your own pace
- Spend < 20 min on each worksheet

2A Qts: · /2 , scikit ,