Homework 3

10-405/10-605: Machine Learning with Large Datasets

Due Monday, March 2nd at 9:00 AM

1 Introduction

This assignment involves understanding basics of building machine learning pipelines and techniques in training logistic regression models. And use principal component analysis(PCA) and feature-based aggregation for exploratory data analysis.

This assignment consists of two major parts. The first part is to build a machine learning pipeline to apply a logistic regression for prediction on a read-world dataset. The second part is to apply PCA to a light-sheet imaging dataset and complete a feature-based aggregation to find a moving visual pattern of neural activity.

2 Logistics

We provide the code template for this assignment in a Jupyter notebook. What you need to do is to follow the instructions in the notebook and implement the missing parts marked with '`<FILL IN>`' or '`# YOUR CODE HERE`'. Most of the '`<FILL IN>/YOUR CODE HERE`' sections can be implemented in just one or two lines of code.

2.1 Getting lab files

You can obtain the notebook '`assignment_notebook.ipynb`' after downloading and unzipping `hw3.zip` at here.

Next, import the notebook into your Databricks, which provides you a well-configured Spark environment and will definitely save your time (see the next section for details).

2.2 Preparing for submission

We provide several public tests via `assert` in the notebook. You may want to pass all those tests before submitting your homework.

In order to enable auto-grading, please do not change any function signatures (e.g., function name, parameters, etc) or delete any cells. If you do delete any of the provided cells (even if you re-add them), the autograder will fail to grade your homework. If you do this, you will need to re-download the empty 'assignment_notebook.ipynb' file and fill in your answers again and resubmit.

2.3 Submission

1. Export your solution notebook as a IPython notebook file on Databricks via `File -> Export -> IPython Notebook`
2. Submit your solution via Gradescope (Please don't rename your notebook file).

3 Setting up environments on Databricks

We provide step-by-step instructions on how to configure your Databricks platform. We also introduced it in detail during the recitation on January 17. The recitations slides can be found here.

1. Sign up for the **Community Edition** of Databricks here: https://databricks.com/try-databricks.
2. Import the notebook file we provide on your homepage: `Workspace -> Users -> Import`
3. Installing third-party packages that will be used in the homework on Databricks: `Workspace -> Create -> Library`. Then select `PyPI`, enter the package name as `nose`. Finally click `Create` to install it. You may also want to check `install automatically on all clusters` option to install the PyPI package on all clusters you create.
4. Create a cluster: `Clusters -> Create Cluster`. You can use any cluster name as you like. When configuring your cluster, make sure to choose **the default Databricks runtime version and Python version** `Python 3`. Note: It may take a while to launch the cluster, please wait for its status to turn to '`active`' before start running.
5. You can start to play with the notebook now!

Note: Databricks Community Edition only allows you to launch one 'cluster'. If the current cluster is 'terminated', then you can either (1) delete it, and then create a new one, or (2) activate and attach to the existing cluster when running the notebook.

4 Logistics Regression

In this section you will go through the steps for creating a click-through rate (CTR) prediction pipeline. You will work with the a sample data that we sampled from Criteo Labs dataset. You will use logistic regression by constructing your own one-hot-encoding(OHE).

This excercise covers:

- **Part 1: Featurize categorical data using one-hot-encoding (OHE)**
- **Part 2: Construct an OHE dictionary**
- **Part 3: Parse CTR data and generate OHE features**
- **Part 4: CTR prediction and logloss evaluation**
- **Part 5: Reduce feature dimension via feature hashing**

Our goal is to accurately predict power output given a set of environmental readings from various sensors in a natural gas-fired power generation plant. In this exercise, we are going to go through a general machine learning pipeline, including data preparation, data modeling, and tuning and evaluation.

5 Principle Component Analysis (PCA)

This section delves into exploratory analysis of neuroscience data, specifically using principal component analysis (PCA) and feature-based aggregation. We will use a dataset of light-sheet imaging recorded by the Ahrens Lab at Janelia Research Campus.

Our dataset is generated by studying the movement of a larval zebrafish, an animal that is especially useful in neuroscience because it is transparent, making it possible to record activity over its entire brain using a technique called light-sheet microscopy. Specifically, we'll work with time-varying images containing patterns of the zebrafish's neural activity as it is presented with a moving visual pattern. Different stimuli induce different patterns across the brain, and we can use exploratory analyses to identify these patterns.

In this section you will learn about PCA, and then compare and contrast different exploratory analyses of the same data set to identify which neural patterns they best highlight.

The section covers:

- **Part 1: Work through the steps of PCA on a sample dataset**
- **Part 2: Write a PCA function and evaluate PCA on sample datasets**
- **Part 3: Parse, inspect, and preprocess neuroscience data then perform PCA**
- **Part 4: Feature-based aggregation and PCA**

See the notebook for detailed descriptions and instructions of each question.