Project Title: Cancer Subtype Multi-Class Classification in Gene Expression Data

Short Description: This project deals with a multi-class classification task (5 tumor types) within RNA-seq gene expression data. This project will introduce participants to topics such as data cleaning, clustering, feature selection methods, and machine learning modeling.

Suggested Tags: multi-class, classification, machine learning, feature selection, automl

Detailed Description:

This project examines gene expression data first made available by the TCGA Pan Cancer analysis project, and included in the UCI Machine Learning Repository, including a multi-class outcome. This outcome is the 'dependent variable' that we want to be able to accurately predict using a trained model. The source data has been sampled so that there is enough data for us to train models, without being too bulky to easily store locally or in the cloud. In total it includes 801 instances (i.e. samples/rows), and 20,531 gene expression features (i.e. independent variables/columns), and 5 cancer subtypes as the multi-class outcome (with class imbalance). Given that this is real-world data it is uncertain how many predictive vs. non-predictive features are in this dataset, however previous analysis of this dataset indicates that models can be trained on this data with high predictive accuracy.

The primary goals of this project are to (1) utilize unsupervised learning approaches to explore relationships between gene expression features in this data, and (2) utilize supervised learning approaches to classify cancer type based on the gene expression features.

Unsupervised learning approaches could, for example, include dimensionality reduction (such as principal component analysis), and clustering (comparing instances in discovered clusters to the true class labels of cancer subtypes.

Supervised learning approaches could, for example, include applying multi-class machine learning modeling with any number of potential algorithms (e.g. decision tree, random forests, support vector machines, artificial neural networks, etc), as well as strategies for feature learning and feature selection.

Ultimately the challenge is to assemble unsupervised and/or supervised learning elements into an analysis pipeline to examine relationships between features and outcome in order to achieve the best prediction performance possible. In other words: What ML algorithm or algorithm(s) perform best? Also, what is the most effective way to set up a data analysis/machine learning pipeline that adheres to best practices in data science?

A secondary goal of this project is to seek to explain or interpret the models or findings from this analysis (e.g. model feature importance estimation). In other words: What gene expression features are most important for driving accurate model performance, vs. features likely to be non-predictive or include redundant information?

As a simpler starting point, as well as to be able to try out the Automated Machine Learning Tool, 'STREAMLINE' (which as of 3/14/24 only handles binary classification), this gene expression dataset could also be encoded as a binary classification problem, with BRCA encoded as class 0 and all other

classes encoded as class 1. This effectively changes the classification task to ask: How well can the BRCA cancer subtype be distinguished from the other 4?

Further, while the gene expression features in the included dataset are given dummy names the original probe names are available through external sites, which can allow exploration of the biological relevance of the findings. Analyses such as gene-set enrichment could be conducted, for example.

Abstract from the UCI Dataset Repository

This collection of data is part of the RNA-Seq (HiSeq) PANCAN data set, it is a random extraction of gene expressions of patients having different types of tumor: BRCA, KIRC, COAD, LUAD and PRAD.

Abstract from the Original Publication (Weinstein et. al. 2013):

The Cancer Genome Atlas (TCGA) Research Network has profiled and analyzed large numbers of human tumors to discover molecular aberrations at the DNA, RNA, protein and epigenetic levels. The resulting rich data provide a major opportunity to develop an integrated picture of commonalities, differences and emergent themes across tumor lineages. The Pan-Cancer initiative compares the first 12 tumor types profiled by TCGA. Analysis of the molecular aberrations and their functional roles across tumor types will teach us how to extend therapies effective in one cancer type to others with a similar genomic profile.

Dataset Sources:

Dataset files are included in the folder 'data'. Once this project is downloaded and unzipped, these datasets are ready to work with.

These data files are alternatively available from the UCI repository:
https://archive.ics.uci.edu/dataset/401/gene+expression+cancer+rna+seq

From the UCI repository the data is downloaded as the file: TCGA-PANCAN-HiSeq-801x20531.tar

To work with this alternative file download, first extract the dataset from the '.tar' file. For example, you can unzip the .tar file, as well as secondary .tar file that is unpacked to get the two underlying data files ('data.csv', and 'labels.csv').

The original data in its entirety can also be downloaded. This dataset includes original probe names, but is a good deal larger and more difficult to work with: https://www.synapse.org/#!Synapse:syn4301332

Dataset Information:

'data.csv' includes the 'X' part of the dataset (i.e. rows of instances, identified by sample identifier, and columns of gene expression features)

'labels' includes the 'y' part of the dataset (i.e. rows of class-outcome, also identified by the same sample identifier)

It may be useful to start by merging these two elements together into a single dataset or dataframe (in pandas).

Samples (instances) are stored row-wise. Variables (features) of each sample are RNA-Seq gene expression levels measured by illumina HiSeq platform. There are no missing values.

A dummy name (gene_XX) is given to each feature. Check the original data from www.synapse.org or the platform specs for the complete list of probes name. The features are ordered consistently with the original data.

Cancer Subtypes in the Dataset (Multi-class outcomes)

- BRCA: Breast Invasive Carcinoma
- KIRC: Kidney Renal Clear Cell Carcinoma
- COAD: Colon Adenocarcinoma
- LUAD: Lung Adenocarcinoma
- PRAD: Prostate Adenocarcinoma

Getting Started Guide:

1. Download the zipped project file, and unzip it's contents to view: a pdf of this project summary, an example Jupyter Notebook, an html link to view the example notebook without having Jupyter Notebook installed, some relevant papers, and the project dataset files.
2. Familiarize yourself with the goals and included data in this project.
3. Read or skim some of the included 'papers'.
    a. In particular, check out the "Cancer Genome Atlas Pan-Cancer" paper as the source of this dataset.
    b. Both 2018 papers, explain the challenge of epistatic interactions in data as well as focus on feature selection strategies that are sensitive to interactions as well as genetic heterogeneity.
    c. The multiclass cancer diagnosis paper gives one example of using machine learning in a multiclass analysis of gene expression data in cancer (but on a different dataset than the one included in this project.
4. (For Beginners) Take some time to learn the basics of Python programming, and using Jupyter Notebook and/or Google Colab Notebooks (see the 'Resources' page for educational links).
    a. Start by installing Anaconda – which comes with both Python, Jupyter Notebook, and all standard Python packages used in ML/data science (e.g. pandas, numpy, scikit-learn, etc.)
    b. Learn to open up and start working with code in Jupyter Notebook and/or Google Colab Notebook.

    c. Take some time to learn some of the basics of data science and machine learning. One starting point would be this YouTube tutorial we created titled "Machine Learning Essentials for Biomedical Data Science"
5. Open up (and run) the included 'Example_Jupyter_Notebook.ipynb' file using Jupyter Notebook. This will both demonstrate that you have anaconda and jupyter notebook installed correctly, as well as provide an example of loading the 'data' and 'lables' files, merging them into a single dataset, conducting a brief exploratory analysis, applying principle component analysis and K-means clustering and then training and evaluating decision tree and random forest machine learning models.
6. Start playing around with this working example notebook as a starting point. Try out making changes to the code: for example, changing the appearance of plots, adding new elements to the exploratory analysis, or changing the machine learning algorithm(s) used to train a model.
7. (Optional) Try out the 'STREAMLINE' Automated Machine Learning software located on the STREAMLINE repository on GitHub.
    a. View the STREAMLINE video tutorials on what it is and how it works.
    b. Before running STREAMLINE, write some code to re-encode the 'Class' column of the data as a binary outcome (e.g. BRCA vs. all other classes)
    c. Download the repository and open up the included demonstration Jupyter Notebook, and update the notebook (as indicated in the notebook instructions) to load and analyze the binary-class encoded dataset in a folder on it's own. You can update the STREAMLINE run parameters directly in this notebook as desired.
    d. Review the STEAMLINE output PDF and folder for a comprehensive ML analysis of these datasets.
8. Lay out some initial specific goals/objectives for yourself and/or your team, regarding what you want to accomplish in building your own analysis pipeline, or utilizing STREAMLINE as a starting point to build from.
9. Try out and compare new strategies, methods, tools, AutoML's to answer your own questions and goals, using the goals at the beginning of this project description as a guide.

References and Suggested Reading:

Original Publication

- Weinstein, J.N., Collisson, E.A., Mills, G.B., Shaw, K.R., Ozenberger, B.A., Ellrott, K., Shmulevich, I., Sander, C. and Stuart, J.M., 2013. The cancer genome atlas pan-cancer analysis project. Nature genetics, 45(10), pp.1113-1120.

Other Useful Publications

- Urbanowicz, R.J., Meeker, M., La Cava, W., Olson, R.S. and Moore, J.H., 2018. Relief-based feature selection: Introduction and review. *Journal of biomedical informatics*, *85*, pp.189-203.
- Urbanowicz, R.J., Olson, R.S., Schmitt, P., Meeker, M. and Moore, J.H., 2018. Benchmarking relief-based feature selection methods for bioinformatics data mining. *Journal of biomedical informatics*, *85*, pp.168-188.
- Olson, R.S. and Moore, J.H., 2016, December. TPOT: A tree-based pipeline optimization tool for automating machine learning. In Workshop on automatic machine learning (pp. 66-74). PMLR.

- Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C.H., Angelo, M., Ladd, C., Reich, M., Latulippe, E., Mesirov, J.P. and Poggio, T., 2001. Multiclass cancer diagnosis using tumor gene expression signatures. Proceedings of the National Academy of Sciences, 98(26), pp.15149-15154.
- Ooi, C.H. and Tan, P., 2003. Genetic algorithms applied to multi-class prediction for the analysis of gene expression data. Bioinformatics, 19(1), pp.37-44.
- Senbagamalar, L. and Logeswari, S., 2024. Genetic Clustering Algorithm-Based Feature Selection and Divergent Random Forest for Multiclass Cancer Classification Using Gene Expression Data. International Journal of Computational Intelligence Systems, 17(1), p.23.
- Urbanowicz, R., Zhang, R., Cui, Y. and Suri, P., 2023. STREAMLINE: A Simple, Transparent, End-To-End Automated Machine Learning Pipeline Facilitating Data Analysis and Algorithm Comparison. In Genetic Programming Theory and Practice XIX (pp. 201-231). Singapore: Springer Nature Singapore.

Project Author:

Ryan Urbanowicz, PhD (he/him) - (Project based on UCI Repository indicated above)

Assistant Professor of Computational Biomedicine at the Cedars Sinai Medical Center

Adjunct Assistant Professor of Biostatistics, Epidemiology, and Informatics at the University of Pennsylvania

Director of Cedars AI Campus Program

Director of the URBS-Lab

URBS-Lab YouTube Channel

URBS-Lab GitHub

Twitter(X)

LinkedIn