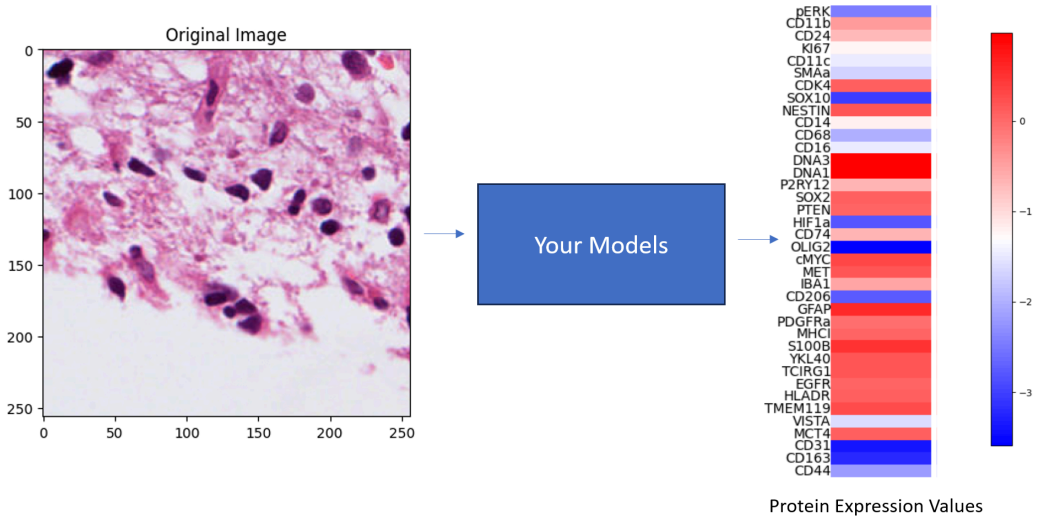
Data Mining 2024 Assignment 2: Prediction of Protein Expression

(by Fayyaz Minhas)

Version: 13 Mar 2024 (Minor clarifications added from the original 16 Feb Version, highlighted in blue font)

In this assignment, the objective is to develop regression models for predicting the level of expression of different proteins in a given biological tissue image. You do not need to know any biology for solving this machine learning exercise. **Your task is to develop machine learning models that use training data (images with known protein expression values) to predict protein expression in test images. The protein expression of different proteins in a given image patch is called its protein expression profile.**



Data Availability, Format and Reading:

The data comes from 4 biological tissue specimens (labelled A1, B1, C1 and D1) and each specimen contains multiple “spots”. Each spot corresponds to a spatial location in a specimen. For each spot, we have an image of that location in the specimen in the form of a png file and the corresponding expression values of 38 different proteins. The total number of spots across all 4 specimens is 9,921.

Instructions on how to download the data and view it (along with other helpful hints) are given in the notebook:
https://github.com/foxtrotmike/CS909/blob/master/DataMining2024_assignment_2.ipynb

Training and Testing:

Unless otherwise specified, use (all or some) data from specimens **A1, B1 and D1 for training and validation** (it is entirely upto you how much data you use for training and how much data you use for validation) and the data for specimen **C1 for testing**. Do not perform testing on image C1 until you have developed your model fully. Also do not use any data from C1 in training or model selection or hyperparameter optimization. Wherever applicable, performance metrics for the test data are to be reported unless otherwise specified. **Note** that Q3ii asks you to report "Leave one specimen out" cross-validation results.

Submission: You are expected to submit a **single Python Notebook** containing all answers and code. Include all prediction metrics in a presentable form within your notebook and include the output of the execution of all cells in the notebook as well so that the markers can verify your output. **Also submit a consolidated table of your performance metrics within the notebook to indicate which model performs the best (MANDATORY).**

Use of additional libraries: You can use other libraries where needed. But please include the installation instructions of those in the notebook along with a reason why you needed to use them.

Use of additional data: You can use other datasets if you want. Please explain any such uses clearly in your notebook.You are free to do any augmentations or any other strategies to improve prediction performance as long as you do not use target variable information directly or indirectly in doing so.

Restrictions:

Students are restricted from sharing the data files or the assignment solutions. Each student needs to submit a single solution which should be developed by the student without assistance from other sources.

Optional Background Reading:

Understanding the importance of spatial protein expression profiling is useful for developing effective regression models in the context of biological tissue analysis, even if you're approaching this from a purely technical perspective as a computer scientist. Here's why this task is both significant and challenging:

1. Bridging Biology with Technology: Spatial protein expression profiling is a technique that maps the location and concentration of proteins within biological tissues. This information is pivotal because proteins are the workhorses of cells, involved in virtually every cell function. Their spatial distribution can reveal insights into cellular processes, disease mechanisms, and

potential therapeutic targets. As a computer scientist, you are essentially translating biological signals into quantifiable data that can be analyzed and predicted with machine learning models. This bridge between biology and technology enables advancements in personalized medicine, drug development, and our understanding of complex diseases.

2. Complexity of Biological Data: Biological tissues are inherently complex, composed of various cell types each expressing thousands of proteins in distinct patterns. The challenge lies in accurately capturing and quantifying this complexity from image data. Your regression models will need to discern subtle variations in protein expression across different tissue regions, which can be critical for diagnosing diseases or understanding tissue function. The ability to predict protein expression profiles from new images can significantly accelerate biological research and medical diagnostics.

3. Personalized Medicine: Different individuals can exhibit unique protein expression patterns, even in the context of the same disease. By developing models that accurately predict protein expression, you contribute to the foundation of personalized medicine, where treatments can be tailored based on an individual's specific protein expression profile. This approach can lead to more effective therapies with fewer side effects.

4. The Challenge of High-Dimensional Data: Image-based protein expression data is high-dimensional, with each pixel or image patch representing multiple features (e.g., color channels, texture, shape). Your task involves extracting meaningful patterns from these features to predict the expression levels of various proteins. This requires sophisticated feature extraction and regression techniques capable of handling the complexity and variability of the data.

5. Machine Learning as a Discovery Tool: Beyond prediction, your models can also uncover previously unrecognized patterns in protein expression, potentially leading to new biological insights. By identifying correlations and trends in the data, machine learning can suggest hypotheses for further biological investigation, making it a powerful tool for discovery.

Question No. 1: (Data Analysis) [15 Marks]

Using training data, answer the following questions:

i. Counting Examples: Determine the number of "examples" or spots present in each specimen. [2 marks]

ii. Protein Expression Histograms: For each specimen, generate histograms to visualize the expression values of 'NESTIN', 'cMYC', and 'MET' and discuss your observations. [3 marks]

iii. Image Pre-processing: Convert a selection of images from RGB to HED color space, focusing on the Hematoxylin channel (H) to highlight cellular nuclei. Provide visual examples and follow the hints in the provided notebook. [4 marks]

iv. H-channel Analysis: Calculate the average intensity of the H-channel for each image. Create a scatter plot comparing these averages against the expression levels of NESTIN for each image. Assess the correlation between H-channel intensity and NESTIN expression. Discuss the potential of H-channel average as a predictive feature for NESTIN expression. [3 marks]

v. Performance Metrics for Prediction: Discuss suitable performance metrics for predicting protein expression from images. Identify the most appropriate metric for this specific problem and justify your choice. [3 marks]

Question No. 2: (Feature Extraction and Classical Regression) [40 Marks]

For the following questions, use the expression of NESTIN as the output prediction target variable.

i) [20 Marks]

Extract features from an image, focusing on the following aspects:

1. Calculate the average and variance for each of the 'H' (from HED), red, green, and blue channels.
2. Additionally, consider other potentially useful features for this task and justify their inclusion. For example you can use one or more of the following features:
    a. PCA (Principal Component Analysis): Applying PCA, such as randomized PCA or incremental PCA, can significantly reduce dimensionality while preserving the variance in the image data, making it easier to identify patterns. This is particularly useful for large datasets or high-resolution images where computational efficiency is a concern. Refer to sklearn.decomposition.PCA for implementation details. You might choose to reduce the dataset size or image dimensions for PCA to manage computational complexity.
    b. GLCM (Gray Level Co-occurrence Matrix): GLCM features can provide insights into the texture of the image, capturing aspects like contrast, correlation, and homogeneity, which might be relevant for distinguishing between different image types. See scikit-image GLCM features for more information.
    c. Transfer Learning Features: Utilizing a pre-trained neural network to extract feature embeddings can leverage learned patterns from vast datasets, potentially improving your model's ability to generalize from the visual content of the images.

Feature Evaluation:

- After feature extraction, plot scatter plots and calculate the correlation coefficient for each feature (from steps a-c) versus the target variable. This analysis will help identify which features are most predictive of the target.
- Important Features: Discuss the importance of selected features based on their correlation with the target variable and their contribution to model performance.

Note: Ensure that PCA and any model fitting only use training data to avoid information leakage from the test set. You can also consider resizing images or selecting specific image regions or reducing the number of training images if necessary to manage computational load.

ii) [20 Marks]

Apply the following regression models using the features from Q2(i):

- Ordinary Least Squares (OLS) Regression or Multi-layer Perceptron (MLP) (your choice!)
- Support Vector Regression (SVR)

For each model, create scatter plots to compare the true and predicted values on the test data. Additionally, evaluate and report your models' performance using the following metrics: RMSE, Pearson Correlation Coefficient, Spearman Correlation Coefficient, and R2 score. Reference for metrics: sklearn.metrics.

You may choose either sklearn or PyTorch for implementation. It's your responsibility to select appropriate hyperparameters, such as the kernel and its parameters for SVR, or the architecture specifics for the MLP.

Deliverables:

Scatter plots for true vs. predicted values for each model type.

Performance metrics (RMSE, Pearson, Spearman, R2 score) on the test data.

Question No. 3 (Using Convolutional Neural Networks) [45 Marks]

(i) [20 Marks]

Develop a Convolutional Neural Network (CNN) using PyTorch to predict the expression level of NESTIN from input images, following the approach outlined in part (ii) of Question (2). Design the architecture of the CNN to input an image and output a single value representing the NESTIN expression level. You have the freedom to select the structure of the network and the

loss functions to be used. You can use pre-trained models and perform transfer learning if needed.

Evaluate your model's performance on the test dataset by creating a scatter plot that compares the true vs. predicted NESTIN expression values. Additionally, quantify your model's accuracy using the following metrics:

- RMSE (Root Mean Square Error)
- Pearson Correlation Coefficient
- Spearman Correlation Coefficient
- R2 score

Your model will be assessed based on its architecture design and the achieved performance metrics. Aim for the best possible performance on the test set, ensuring that the test data is not used during training. Include in your submission convergence plots that illustrate the change in loss across training epochs, demonstrating how your model's performance improves over time.

(ii) [20 Marks]

Create a neural network using PyTorch to simultaneously predict the expression levels of five specific proteins (EGFR, PTEN, NESTIN, SOX2, and MET) from given image patches. You have the flexibility to choose the architecture of the neural network and the loss functions you deem appropriate for this task.

For model validation, employ a "leave one specimen out cross-validation" strategy. This approach involves sequentially using data from one specimen as the test set and the combined data from the remaining specimens as the training set. This method is similar to a 4-fold cross-validation but specifically tailored to ensure that each specimen is used as a test set exactly once. This validation technique ensures that your model's performance is evaluated on entirely unseen data, mimicking a scenario where the model is tested on data from a new specimen. For a practical understanding of how this is implemented, you can refer to the GroupKFold method in scikit-learn.

After training and validating your model, create a total of 20 scatter plots to visually represent your results. These plots should compare the predicted versus actual expression levels for each of the five proteins across the four specimens. This means you'll generate one plot for each protein per specimen.

Finally, quantify the performance of your optimal model for each protein with the following statistical metrics:

- RMSE (Root Mean Square Error): Measures the model's prediction error.
- Pearson Correlation Coefficient: Assesses the linear relationship between predicted and actual values.
- Spearman Correlation Coefficient: Evaluates the monotonic relationship between predicted and actual values.

- R2 Score: Indicates the proportion of variance in the dependent variable predictable from the independent variable(s).

For each metric, report both the average and standard deviation across the specimens for every target protein. This comprehensive evaluation will help in understanding the model's predictive accuracy, reliability, and the nature of its errors or biases.

HINT:

A naïve (but possibly effective) strategy can be to simply try the same network architecture in 3(a) and train different models separately for each target variable.

iii) [5 Marks] Discuss limitations and possible extensions of the optimal pipeline, e.g., is there any additional information we can utilize to improve prediction performance and how that can be used? You will be graded on the feasibility and practicality of your ideas and you can get bonus marks depending upon whether you show any preliminary or pilot results. [5 Marks Max]