Detecting Hypoxia with Machine Learning

Florian Daefler, Arianna Feliziani, Federico Russo, Jan Schrewe, Changchen Yu
Bocconi University

Abstract
As cancer cells rapidly proliferate, they often outgrow their blood supply, resulting in regions of hypoxia (low oxygen) within the tumor microenvironment. Hypoxic conditions promote tumor aggressiveness, resistance to therapy, and the generation of genetic alterations that drive cancer progression. The aim of this report is to develop a robust method for detecting hypoxia and normoxia in cells using gene expression analysis and Machine Learning classification models.

Contents

1 Data Analysis

1.1 Data

The analysis is based on 2 datasets, SmartSeq and DropSeq, corresponding to different sequencing methods. They contain the frequencies of genes under several experiments, with different cell lines, positions of sequencing reaction and oxygen conditions. Each cell belongs to one of 2 cell lines, MCF7 or HCC1806, which we analyse separately as they exhibit structural differences in terms of genes expression. The binary target variable is the oxygen condition, which indicates either normal oxygen levels (NORMOXIA) or low oxygen levels (HYPOXIA). To sum up, there is a total of 4 datasets: SmartSeq MCF7, SmartSeq HCC1806, DropSeq MCF7, DropSeq HCC1806. In the following sections, we will show some of the data analysis results only on the SmartSeq datasets, but all the learning models are run both on SmartSeq and DropSeq datasets.

Cell line	SmartSeq	DropSeq
MCF7	(383, 22943)	(21626, 3000)
HCC1806	(243, 23405)	(14682, 3000)

Table 1: Dimensions of the 4 datasets (number of cells and genes).

From the table above we note that in the SmartSeq dataset the number of genes vastly exceeds the one of cells, which poses a problem for classic machine learning problems, prone to overfit in high-dimensional settings. After transforming the datasets, we obtain the following structure: each cell is a data point, with gene frequencies as features and the oxygen condition as target variable.

Experiment	WASH7P	MTND1P23	MTND2P28	...	LABEL
NormoxiaS123	0	250	54	...	0
HypoxiaS97	0	11	0	...	1

Table 2: Example of the dataset structure.

We do not detect a significant class imbalance in the datasets. Moreover, we run some analyses on the distributions of both cells and genes of the SmartSeq datasets, that is the distributions of the gene expression in one cell and the one of single genes across different cells), which we both found highly non-normal and skewed. In every cell only at most 50% of the genes are expressed, but this is not the case when we analyze genes instead of cells. There are indeed both genes that appear very often and very rarely; a possible explanation is that those that appear very often are the genes whose function is necessary to keep the cell alive.

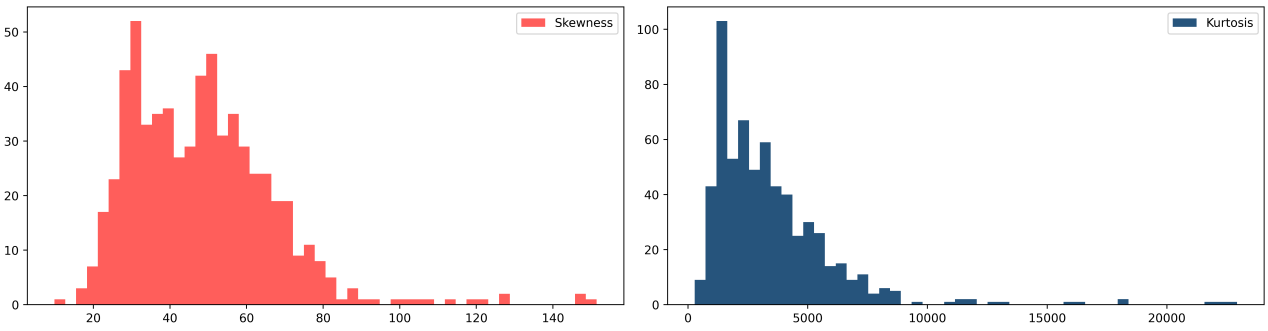

Figure 1: Histograms of skewness (left) and kurtosis (right) of cells distributions.

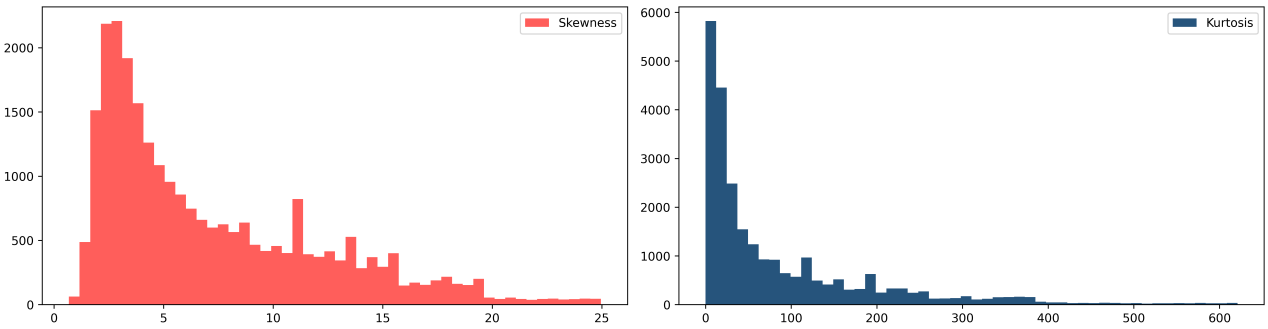

Figure 2: Histograms of skewness (left) and kurtosis (right) of genes distributions.

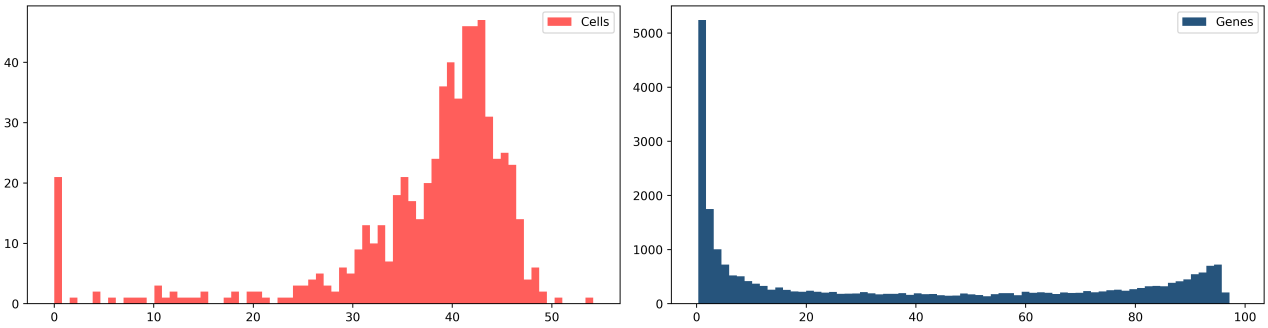

Figure 3: Percentage of nonzero values on cells (left) and genes (right) distributions.

We also implemented an interactive graph, which helped getting a better understanding of the data provided. In the interactive graph, one is able to select a specific gene and select one graph among two different stacked bar charts. The first one represents the value of the genes grouped by the condition of the cell it appears in, while the second one is a normalized version of the first and allows us to quickly understand, given a certain value of the gene expression, the proportion in which such value is attained in normoxic and hypoxic cells.

As it is clear from the graphs below, the MTCO1P12 gene has a more balanced frequency in normoxic and hypoxic cells, while when the NDRG1 gene appears more than 2352 times the cells are only hypoxic. From this, of course, we cannot draw conclusions on the data or on the relation between gene appearance and the target variable, however it is interesting to visualize this information to get a better understanding of the analysis that we are pursuing.

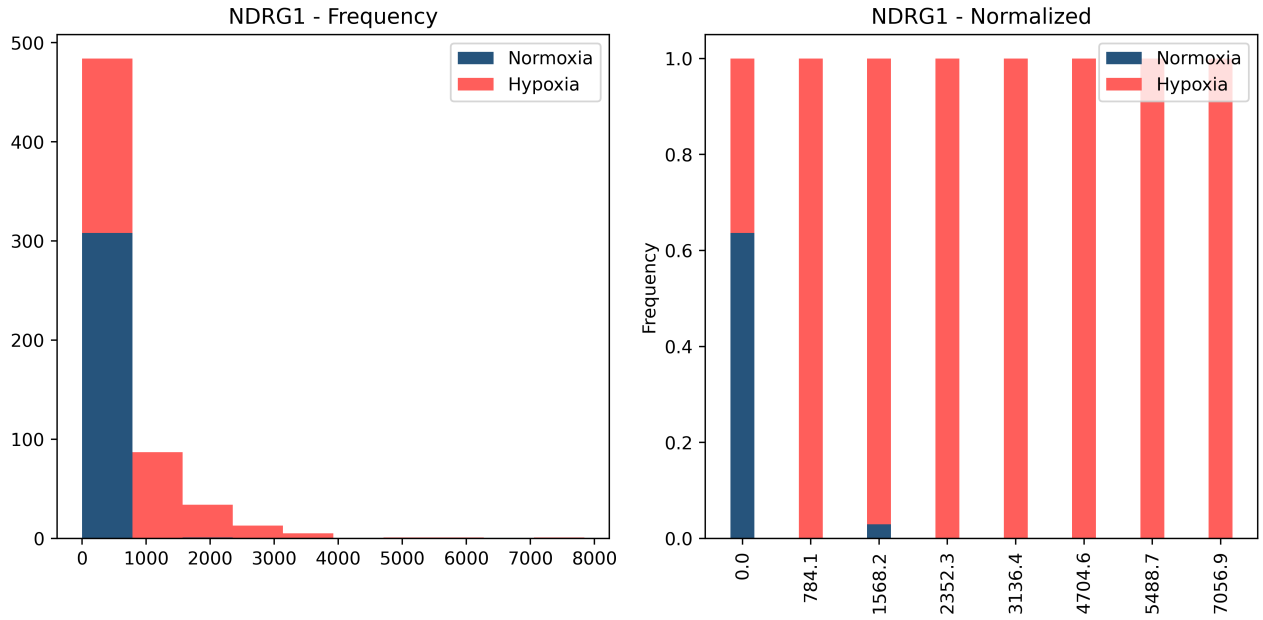

Figure 4: Histogram (left) and normalized histogram (right) of NDRG1 expression.

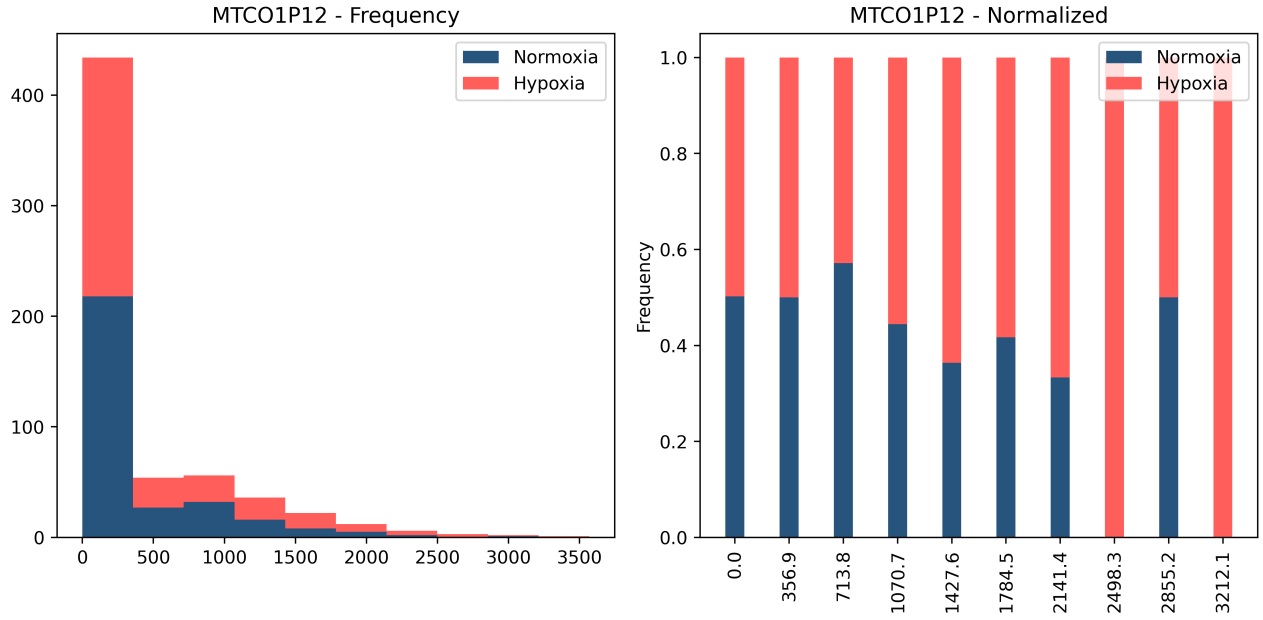

Figure 5: Histogram (left) and normalized histogram (right) of MTCO1P12 expression.

1.2 Dimensionality Reduction

For the reasons seen above, we are interested in reducing the dimensionality of the data. The first technique that we employ is Principal Component Analysis (PCA), which reduces original features to their principal components (PCs), that is linear combinations, while maximizing the variance of the compressed data. In particular, we seek to retain 95% of the variance, which requires 20 principal components for SmartSeq MCF7 and 34 for SmartSeq HCC1806. Below you can find the results of the PCA on SmartSeq datasets.

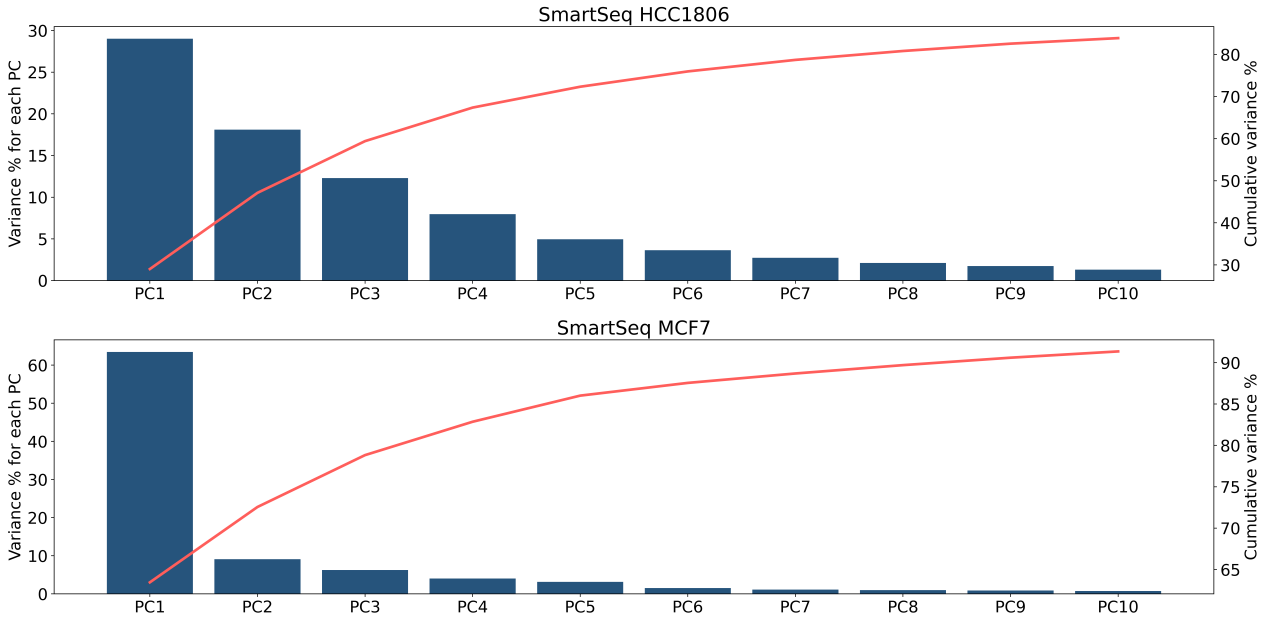

Figure 6: Variance explained by each PC of the PCA.

While for MCF7 we clearly observe that the first PC is more important than the subsequent ones, for HCC1806 also the secondary PCs manage to explain a significant part of the variance.

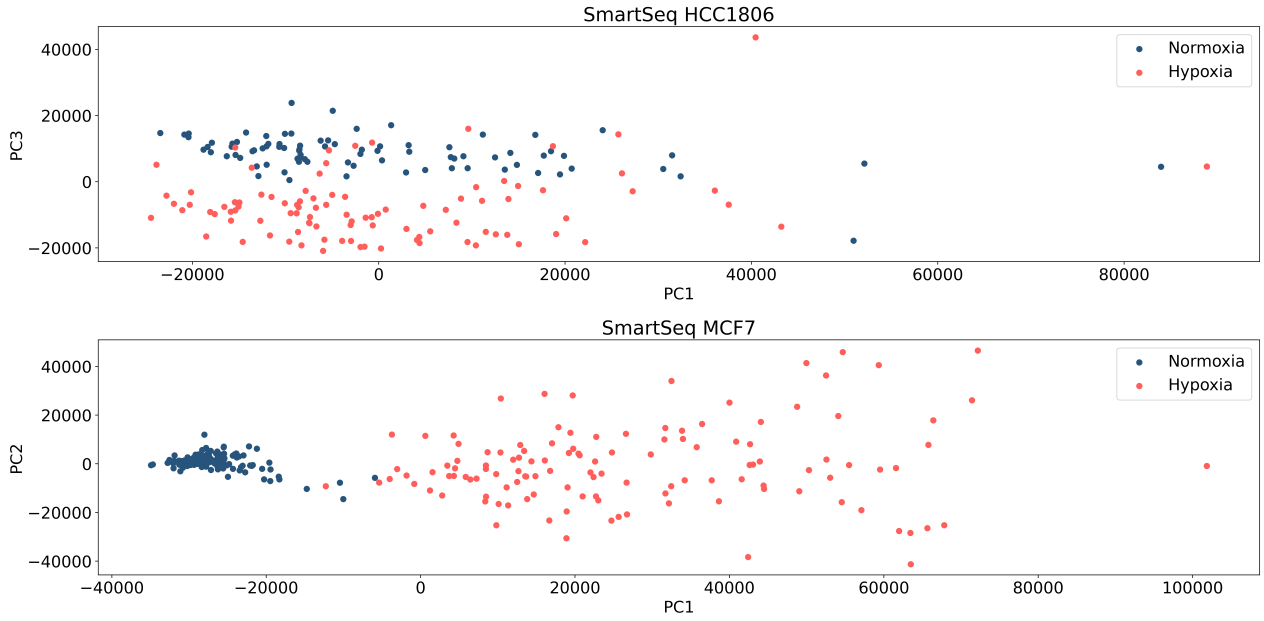

Figure 7: Value of PCs and the target variable.

The chart above shows that PCs are helpful in discriminating between NORMOXIA and HYPOXIA. In particular, for HCC1806 it's mainly the third PC, while for MCF7 it's the first.

2 Clustering and Cell Cycle Analysis

2.1 Clustering Algorithms

Clustering is a fundamental building block of unsupervised learning as it can potentially uncover hidden relationship between cells by analyzing the cluster they are assigned to. We will build upon the knowledge acquired in the previous sections and answer the following two questions:

1. Are clustering algorithms able to cluster cells based on their hypoxic condition?
2. If this is not the case, what are some possible interpretation of such clusters?

The clustering is performed exclusively on the SmartSeq dataset.

2.2 K-Means

By using K-means we obtain satisfactory results on the MCF7 cell line, it however struggles to correctly cluster across cell condition those cells belonging to the HCC1806 cell line. This is mainly the result of the PCA data of this cell line not forming meaningful clusters. To solve this problem and answer the second question we move on to more powerful clustering algorithms combined with non-linear dimensionality reduction tecniques.

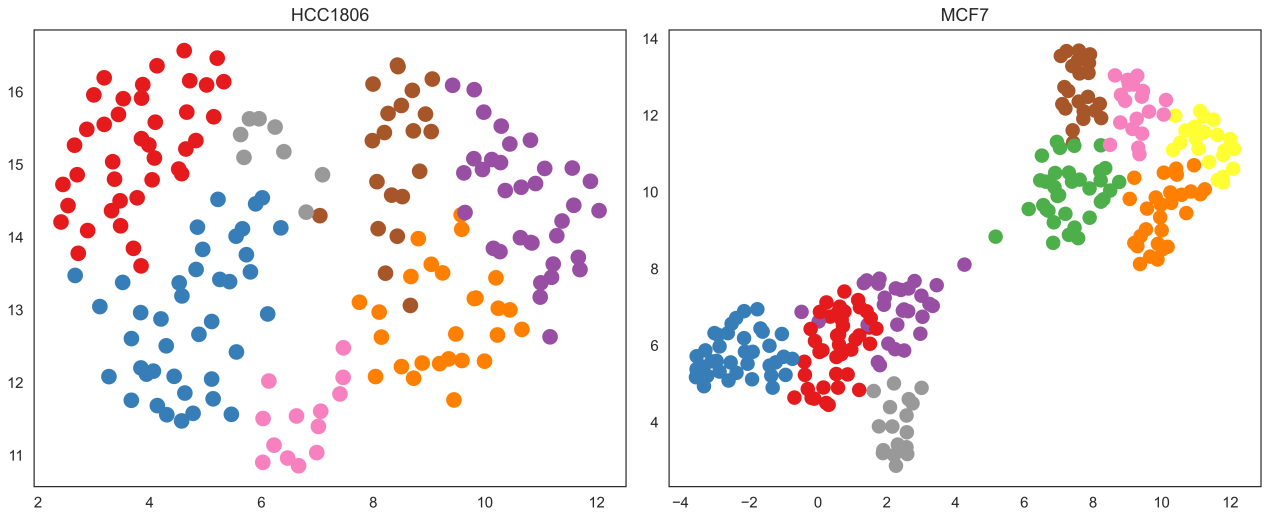

Figure 8: Umap and Leiden algorithms.

2.3 Umap and Leiden Algorithms

In order to identify which cells are undergoing a certan stage of their cell cycle and whether those cells form meaning cluster we selected as our reference some of the genes that score high in the S phase and the G2 phase and analyzed their expression across cells. This analysis is performed using Leiden Algorithm to which we fed the data whose dimensionality was reduced using UMAP.

For the MCF7 dataset, after performing both a qualitative analysis (heatmap) and a quantitative one (violyn plot of the mean expression of selected genes), we conclude that Cluster

1, Cluster 2 and Cluster 3 contain cells undergoing the S phase while Cluster 4 and 6 cells undergoing the G2 phase.

On the other hand, for the HCC1806 dataset we observed an overlap in the clusters between cells that had a high expression of genes of the S and the G2 phase. A possible explanation is that such phenomenon is caused by the presence of a heterogeneous amount of cells undergoing the two phases in the clusters outputted by Leiden Algorithm.

3 Supervised Learning

3.1 Baseline Models

The first step of the supervised learning phase is to test several base models, whose results serve as a baseline for more complex models. The first two models are plain Logistic Regression and Principal Component Regression (PCR), which applies PCA to the features before fitting a Logistic Regression. The results of the 2 regressions show that the Logistic performs better than the PCR on SmartSeq, while the accuracy on the test set of DropSeq is the same for both models.

3.2 Boosting Algorithms

We decided to implement boosting algorithms, which are an instance of supervised learning and ensemble learning, in which different weak classifiers are trained to then build a stronger one. In particular, we implemented AdaBoost, XGBoost and CatBoost. First, we trained the AdaBoost classifier, with which we got an accuracy of 100% on the MCF7 and HCC1806 SmartSeq datasets, and respectively of 98.08% and 95.75% on MCF7 and HCC1806 DropSeq datasets. However, we were not satisfied with the results obtained on DropSeq, specifically on HCC1806. For this reason, we decided to also implement XGBoost, which is another instance of Boosting algorithms that, differently from AdaBoost, supports parallelization. For this reason, it is well suited to handle large datasets, potentially providing more accurate results. As we expected, the results we got on DropSeq with XGBoost are better than the ones with AdaBoost. In particular we achieved an accuracy of 98.55% on MCF7 and of 96.57% on HCC1806, which are among the best results we were able to get throughout our analysis.

3.2.1 Feature Importance

By default, XGBoost estimates feature importance with the gain metric, which calculates the improvement in the model's objective function achieved by splitting on a particular feature. It thus quantifies the contribution of each feature to reducing the loss during the training process.

Catboost uses the so called PredictionDiff to estimate feature importance. This metric measures the impact of changing a particular feature's value on the model's predictions by calculating the absolute difference between the original prediction and the prediction when the feature value is replaced with a missing value. Higher feature importance values indicate that changing the feature's value has a more significant impact on the model's predictions. Analyzing feature importance provides interesting insights into the relationship between hypoxia and gene expression.

- ENO1 GAPDH PGK1 TPI1 are genes in hypoxia hallmark important for MCF7;

- ENO1 IGFBP3 LDHA NDRG1 P4HA1 are genes in hypoxia hallmark important for HCC1806;
- BCYRN1 ENO1 H4C3 MALAT1 RPL35 are retained by both classifiers;
- Catboost considers mitochondrial genes important for predicting hypoxia on MCF7;
- Percentage of non-zero values of a cell is an important feature for both cell lines.

3.3 Ensemble Learning

In our pursuit of pushing the boundaries of prediction performance, we turned to ensemble learning, specifically utilizing Stacking Ensemble. We tested this class of learning algorithms only on the DropSeq datasets, as we were already satisfied by the performance of simpler models on the SmartSeq datasets. During our experimentation, we discovered that applying the same models to both HCC1806 and MCF7 DropSeq datasets yielded subpar results for one of them. As a result, we divided our tuning process into two distinct parts.

For the MCF7 dataset, we identified that employing SVC, XGBoost, and CatBoost, with Logistic Regression as the final estimator, achieved outstanding performance. This combination attained an accuracy of 98.7% on the test dataBy default. On the other hand, when dealing with the HCC1806 dataset, we found that Logistic Regression, SVC, XGBoost, and CatBoost, along with Random Forest as the final estimator, delivered the best results. This configuration achieved a accuracy of 97.2% on the test data.

It is worth mentioning that each addition of new models to the stacking classifier necessitated considerable time for training and testing, typically ranging from one to two hours. Due to the relatively large dataset, this time-consuming process limited our ability to thoroughly explore if these models achieved the optimal performance. Nevertheless, within the given time constraints, these models exhibited superior performance compared to other alternatives.

Model	Smart HCC1806	Smart MCF7	Drop HCC1806	Drop MCF7
Logistic	1.00	1.00	0.95	0.98
PCR	0.96	0.97	0.95	0.98
SVM	1.00	1.00	0.966	0.982
AdaBoost	1.00	1.00	0.958	0.981
XGBoost	1.00	1.00	0.966	0.986
Catboost	1.00	1.00	0.966	0.984
Ensemble	NA	NA	0.972	0.987

Table 3: Test accuracy of all the models.