SVM-Based Copy Number Caller for Human Genomes

Objective

- Primary Goal: The project aims to develop a machine learning model, specifically a Support Vector Machine (SVM), for calling copy number variations (CNVs) in human genomes. CNVs are segments of the genome that have been deleted or duplicated on certain chromosomes. These variations can have significant implications in genetic research and medicine, particularly in the context of genetic disorders and diseases.
- Significance: The accurate detection and classification of CNVs are crucial for understanding genetic variations in populations, diagnosing genetic diseases, and advancing personalized medicine. The project seeks to leverage the power of SVMs to efficiently and accurately identify these variations from genomic data.
- Performance Criteria: Given the critical nature of genomic analysis, the model must demonstrate high accuracy, precision, recall, and F1-score. These metrics ensure that the model reliably identifies true CNVs while minimizing false positives and negatives, which is vital in medical and genetic research applications.
- Novelty: while SVMs have been used for CNV calling before, there is nothing in the literature that applied it to third generation long read sequencing technologies
- Shift from Transformer model: A transformer model was initially proposed for this task, but due to time and compute resource considerations: an SVM approach was better suited and used

Data Loading and Preprocessing

- Data Sources: The data used for this project include genomic markers indicative of CNVs.
- A hidden markov model based CNV caller (*hmcnc*) was used to provide the ground truth set for training. 10 genomes available in the public domain from the Human Pangenome research consortium were used to generate the call sets.
- Initial Processing:
    - The notebook starts with importing necessary libraries such as `csv`, `numpy`, `pandas`, `matplotlib.pyplot`, `seaborn`, and `joblib`. These are crucial for data manipulation, visualization, and model serialization.

        - Function Definition: A function named `load_and_process_bedSVM` is defined. This function is designed for loading and processing genomic data. It specifies columns to use, discards certain columns, and performs grouping and aggregation.
        - Feature Selection: Specific columns relevant to CNV analysis are selected, and unnecessary columns are discarded. This step focuses on retaining only the most informative features for CNV detection.
        - In the end, the coverage and clipping features per 100 bp bins were selected, with padding for variable length sequences.
- Feature Engineering:
    - Encoding and Aggregation: Categorical variables, such as chromosome numbers, are encoded. Data is grouped and aggregated to form more meaningful features for the model.
    - Expansion of Features: Certain columns are expanded into multiple features to capture more nuanced information that might be indicative of CNVs.
- Data Transformation:
    - Scaling: The features are scaled to normalize their ranges. This step is crucial for models like SVMs that are sensitive to the scale of input data.
- Data Exploration and Visualization
    - PCA Visualization: Principal Component Analysis (PCA) is applied for dimensionality reduction, and the results are visualized in both 2D and 3D plots. This is a standard technique to understand the variance in high-dimensional data.
    - t-SNE Visualization: t-Distributed Stochastic Neighbor Embedding (t-SNE) is another technique used for visualizing high-dimensional data in both 2D and 3D. This is particularly useful in genomic data to observe clustering patterns.
    - UMAP Visualization: Uniform Manifold Approximation and Projection (UMAP) is also utilized for data visualization in 2D and 3D. This suggests a thorough exploration of the data structure.

Model Implementation

Selection of the Model

- Model Choice: A Support Vector Machine (SVM) is chosen for this task. SVMs are particularly suited for classification tasks like this due to their effectiveness in handling high-dimensional data and their ability to model non-linear decision boundaries, which can be crucial in complex genomic datasets.

Model Configuration

- Kernel Selection: A Radial Basis Function (RBF) kernel was used as RBF kernels are effective in handling cases where the relationship between class labels and attributes is non-linear.

Model Training

- Data Splitting: The dataset is split into training and test sets. This separation is essential for training the model on one subset of the data and then testing it on an independent subset to evaluate its performance.
- Cross-Validation: Stratified K-Fold cross-validation is used to ensure that each fold of the dataset is a good representative of the whole. It also helps in assessing the model's performance more reliably by using multiple different training and validation sets.

Performance Evaluation

- Training Process: During training, the SVM model learns to distinguish between different classes of CNVs based on the features presented. This process involves finding the hyperplane that best separates the classes in the feature space. Two classes were specified for this model, “0” and “1”.
- “0” indicates a region of the genome with neutral copy state
- “1” indicates a region where there is a deletion or duplication of the genomic content
- Evaluation Metrics: The model is evaluated on metrics like accuracy, precision, recall, and F1-score. These metrics provide a comprehensive view of the model's performance, highlighting its strengths and potential areas for improvement.

Detailed Evaluation of Model Results

The evaluation of the SVM model's results is crucial to understand its effectiveness in calling copy number variations (CNVs) in human genomes. Based on the performance metrics, here is a detailed analysis:

Test Set Evaluation

Accuracy: The model achieves an accuracy of approximately 98.97% on the test set. This high level of accuracy indicates that the model is very effective in correctly classifying both normal and variant genomic sequences.
Precision and Recall by Class:

- Class 0 (Normal): Achieves nearly perfect precision (99%) and recall (100%). This indicates that the model is exceptionally accurate in identifying normal genomic sequences and rarely misclassifies them as variants.
- Class 1 (Variant): Demonstrates high precision (99%) and slightly lower recall (94%). This suggests that while the model is highly accurate when it predicts a variant, it's slightly less consistent in catching all actual variant cases.

F1-Score: The F1-scores are 0.99 for Class 0 and 0.96 for Class 1. The high F1-scores indicate a strong balance between precision and recall for both classes, which is essential in medical and genomic applications where both false positives and false negatives carry significant consequences.

Independent Test Set Evaluation

Accuracy: On the independent test set, the model shows an accuracy of around 97.71%. This consistency in performance on a separate dataset underscores the model's robustness and generalizability.
Precision and Recall by Class: Both classes show high precision and recall (96% to 100%), indicating the model's strong ability to generalize to new, unseen data.
F1-Score: The F1-scores for both classes are around 0.98, maintaining a balanced performance between precision and recall on the independent test set.

Stratified 5-Fold Cross-Validation

Average Metrics: Across the 5-fold cross-validation, the model consistently shows high performance with average accuracy, precision, recall, and F1 score all at 99%. This consistency is indicative of the model’s reliability and its ability to perform well across various subsets of data.

Insights from Model Evaluation

- High Generalizability: The model's consistent performance across both the test set and an independent test set suggests it generalizes well, a key factor in its potential for real-world applications.

- Robust Performance in Variant Detection: The model's ability to detect variants with high precision and recall is crucial, considering the importance of accurate CNV detection in genetic research and diagnostics.
- Balanced Precision and Recall: The high F1-scores across both classes indicate a well-balanced model that doesn’t overly favor either false positives or false negatives. In the context of CNVs, this balance is vital for reliable diagnostics and research findings.
- Stability Across Folds: The uniform high performance in cross-validation highlights the model’s stability and reliability, an important aspect when dealing with complex genomic data.