DATA CLASSIFICATION COURSEWORK REPORT

Student ID: 22018537

Name: Nikunj Bhavsar

Module: 6COM1044-0105-2024 - Machine Learning and Neural Computing

Introduction

Diabetes is a chronic health condition known to affect millions of individuals worldwide. Early detection of diabetes can significantly reduce its impact by facilitating timely intervention and management. This report investigates a machine learning approach to classifying diabetes status using two key techniques: **Principal Component Analysis (PCA)** for dimensionality reduction and data exploration, and **Support Vector Machine (SVM)** classification with a Gaussian Radial Basis Function (RBF) kernel. The data used for this experiment are drawn from a preprocessed set of health indicators, partitioned into four CSV files according to the participants' physical activity levels (NoActivity vs. PhysActivity) and the typical training/test split.

The structure of the report follows four tasks that together constitute a comprehensive machine learning workflow. **Task 1** focuses on data exploration using PCA, **Task 2** builds an SVM model for the NoActivity group, **Task 3** repeats the SVM modeling for the PhysActivity group, and **Task 4** cross-tests each group's model on the other group's data. Each task or subtask includes a **Methodology**, **Results**, and **Discussion** section to document the decisions made, the outcomes observed, and the significance of the findings. By presenting this systematic approach, the report highlights both the strengths of SVM classification for diabetes detection and the complexities inherent in applying a model to groups with potentially different characteristics.

This report has been refined using the Grammarly extension for spelling and sentence grammar refinement.

Data Exploration

Task 1(a). Loading and Splitting Data

Methodology

The experiment began by loading four CSV files: two associated with individuals who did not engage in physical exercise over the past 30 days (referred to as “NoActivity”) and two associated with individuals who did exercise (“PhysActivity”). Each subset consisted of a training set and a test set, leading to four total files for analysis. The code utilized `pd.read_csv()` to load each file into a Pandas DataFrame. Once loaded, a custom function named `split_features_labels(df)` was defined to separate the first column (the binary diabetes label) from the remaining seven columns (the features, which include BMI, GenHlth, MentHlth, PhysHlth, Age, Education, and Income). The features were stored in a variable `X`, while the labels were stored in `y`. This function was called on each DataFrame to ensure consistent handling of all four CSV files.

Results

By the end of Task 1(a), the experiment produced four pairs of variables: `X_noact_train`/`y_noact_train`, `X_noact_test`/`y_noact_test`, `X_phys_train`/`y_phys_train`, and `X_phys_test`/`y_phys_test`. Each pair represented the features and labels for the NoActivity and PhysActivity groups in both training and testing configurations. The inspection of these arrays confirmed that each group contained 701 training samples and 301 test samples. The label distribution was balanced, thus minimizing skew in subsequent classification.

Discussion

Separating features and labels at the outset streamlined the flow of operations, ensuring that any further steps in the pipeline, such as normalization or classification, could focus purely on the features. By confirming the balanced nature of the classes, it was apparent that accuracy metrics would be indicative of the model’s real performance without major concerns about class imbalance.

Task 1(b). Scatter Plots

Methodology.

An initial exploration of the data was undertaken by generating scatter plots to visualize potential relationships between two features: BMI and Age, and the binary label. The code relied on **Matplotlib** and **Seaborn** for plotting. The features to plot were designated as `(BMI, 0)` and `(Age, 4)` based on their column indices. Four subplots were created, two for the NoActivity group (training and test sets) and two for the PhysActivity group (training and test sets). Each point's color was determined by the participant's label (0 or 1), providing a visual hint of how feature values are clustered by diabetes status.

Results.

From the scatter plots, a trend emerged indicating that participants with higher BMI values and older Age tended to be classified as "1" (diabetes or prediabetes). Nonetheless, the overlap between the two classes in each subplot was visible, which hinted that no single pair of features could serve as a perfect separator. In other words, while these two features hold predictive value, additional dimensions likely influence the final classification.

Discussion.

This visualization offered a valuable first look at potential correlations in the dataset. Observing class overlap reinforced the need to use all seven features and possibly apply techniques like PCA and SVM to capture more nuanced patterns. The plots also provided reassurance that the training and test sets appeared similarly distributed for both NoActivity and PhysActivity groups, indicating a consistent data split with no obvious shift.

Task 1(c). Normalization

Methodology.

Before proceeding to reduction or classification, the code standardized each dataset by subtracting the mean and dividing by the standard deviation of its training features. Specifically, a helper function named `normalize_data(train, test)` utilized scikit-learn’s `StandardScaler`. The scaler was fitted on the training set features so that they had mean zero and unit variance, and the same scaling parameters were applied to the test set to ensure consistency. This step was repeated separately for the NoActivity and PhysActivity groups, yielding scaled versions of the training and test features for each.

Results.

Post-normalization, `X_noact_train_scaled` and `X_noact_test_scaled` were ready for subsequent analysis in the NoActivity pipeline, while `X_phys_train_scaled` and `X_phys_test_scaled` were prepared for the PhysActivity pipeline. An inspection of the summary statistics (mean and standard deviation) in each test set revealed that the scaled features were indeed centered near zero, with a spread near one. This outcome confirmed that the normalization process had been conducted as intended.

Discussion.

Normalization is especially important for distance-based algorithms and kernel methods. Features on drastically different scales might cause the classifier to overweight certain attributes. By normalizing each group’s training and test data, subsequent PCA and SVM computations were made more comparable. Additionally, verifying the means and standard deviations in the test sets served as a practical check that no data leakage or scaling mismatch had occurred.

Task 1(d). PCA Analysis

Methodology

Principal Component Analysis (PCA) was performed on each training set (for both NoActivity and PhysActivity) after normalization. The code defined a function `perform_pca(X_train, X_test, n_components=2)` that generated PCA models using two principal components. These components were then plotted in a scatter plot to see whether class 0 and class 1 exhibited separable patterns in a lower-dimensional space. The function also allowed for the extraction of the PCA model itself (`pca_noact` or `pca_phys`) to compute the variance explained by each component if needed.

Results

Visual inspection of the two-dimensional PCA plots showed partial clustering of class 0 vs. class 1 points. Generally, the first principal component accounted for a significant portion of the variance, while the second principal component explained an additional 15–20%. Even combined, these two components did not fully separate diabetic and non-diabetic participants, as overlapping regions remained. This implied that further principal components or the original seven-dimensional space carried additional discriminatory information.

Discussion

Applying PCA uncovered that while BMI, Age, and possibly other health indicators drive a large fraction of the variance, the data's complexity requires more advanced classification methods to achieve a good balance of sensitivity and specificity. The partial overlap in the scatter plots substantiated that linear methods in just two dimensions may not suffice for perfect classification, motivating the subsequent adoption of a non-linear SVM approach.

Task 2: SVM Classification for the NoActivity Group

Methodology

Building upon the scaled NoActivity data, this task created an internal train-validation split from the original training set. Specifically, 70% of `X_noact_train_scaled` became a new training subset, and 30% formed a validation subset. The code used `train_test_split` from scikit-learn, setting a random state of 42 for reproducibility. Next, three potential parameter combinations for the SVM were defined in a list of dictionaries: `[{'C':1, 'gamma':1}, {'C':5, 'gamma':0.5}, {'C':0.5, 'gamma':0.05}]`. An SVM with an RBF kernel was then trained on each parameter set, and performance on the validation set was measured via the `accuracy_score` function.

Results

Out of the tested configurations, `[C=0.5, gamma=0.05]` produced the highest accuracy on the validation subset. With these chosen hyperparameters, the final SVM model was trained on the entire NoActivity training set (701 samples) and tested on `X_noact_test_scaled`. The resulting accuracy (`test_acc`) reached approximately 73.4%. The confusion matrix (`cm`) revealed the counts of true positives, false positives, true negatives, and false negatives, indicating that while many diabetic individuals were correctly identified, a number of misclassifications still occurred.

Discussion

This step validated the usefulness of employing parameter tuning on a separate validation set rather than guessing or relying on default SVM parameters. The moderate success in test accuracy hinted that the model captured meaningful patterns, yet the presence of false negatives and false positives underscored the intricate nature of medical data. While 73.4% accuracy is promising, these results remind practitioners of the potential costs of missing actual diabetic cases (false negatives), suggesting the need to weigh sensitivity and specificity carefully.

Task 3: SVM Classification for the PhysActivity Group

Methodology

A nearly identical procedure was repeated for the PhysActivity group's data. The scaled training set, `X_phys_train_scaled` and `y_phys_train`, was split into 70% training subset (II) and 30% validation using `train_test_split`. The same three parameter combinations for the SVM ([C=1, gamma=1], [C=5, gamma=0.5], and [C=0.5, gamma=0.05]) were tested, this time on the PhysActivity validation samples. Following the validation performance check, the best combination was retrained on all 700 PhysActivity training samples and tested on the 300-sample PhysActivity test set.

Results

This time, `[C=1, gamma=1]` achieved the highest validation accuracy. When evaluated on the PhysActivity test data, the final model yielded an accuracy that hovered around 74.7%. The confusion matrix similarly indicated solid coverage of true positives with a moderate number of false positives and false negatives.

Discussion

The performance improvement, relative to some of the lower or higher parameter choices, suggests that participants who engage in exercise might generate feature distributions that respond well to parameter (C=1) and a moderate gamma (gamma=1). This stands in contrast to the NoActivity group's best settings, which favored C=0.5 and gamma=0.05. The discrepancy emphasizes that even small differences in population behavior can influence the optimal margin and kernel spread. While 75% accuracy is reasonable, the existence of false negatives remains clinically significant. More advanced parameter tuning or feature expansion might further refine these outcomes.

Task 4: Cross-Model Evaluation

Methodology

In the final step, each group's best-performing SVM was evaluated on the other group's test set. To keep data processing consistent, the NoActivity model was first tested on the PhysActivity test data using the **NoActivity** group's `StandardScaler` parameters (that is, the training mean and standard deviation from the NoActivity dataset). The predicted labels were then compared against the actual labels in the PhysActivity test set. Conversely, the PhysActivity model was used to predict the NoActivity test set after scaling those features with the PhysActivity training means and standard deviations.

Results

When the NoActivity model was applied to the PhysActivity test data, the accuracy dropped to around 74%, indicating a notable decrease from the accuracy seen on the NoActivity test set. The PhysActivity model, when tested on NoActivity's test data, yielded approximately 65% accuracy. Although both cross-group accuracies were lower than their respective in-group performances, the NoActivity model seemed to adapt marginally better, with an edge over the PhysActivity model's cross-performance.

Discussion

This outcome underscores the potential differences in feature distributions between physically active and inactive individuals. Despite the models capturing substantial information about each group's health indicators, they did not fully generalize to a different demographic or behavioral pattern. Consequently, for practical applications, developing group-specific models or employing more advanced adaptation strategies might be prudent. This step illuminated that while certain aspects of diabetes risk remain consistent, domain differences (e.g., exercise habits) can shift feature relationships enough to reduce out-of-domain performance.

Conclusion

In conclusion, this experiment demonstrated the effectiveness of combining PCA and SVM for classifying diabetes in individuals. The code systematically processed four CSV files, each representing one subset of a population categorized by physical activity. PCA gave insights into the main dimensions of variance, revealing partial class separation but also highlighting the need for more comprehensive classification methods. The SVM, specifically a C-SVC using the RBF kernel, performed reliably once hyperparameters were selected via a validation set, achieving around 73% accuracy for the NoActivity group and 75% for the PhysActivity group in test scenarios. However, when cross-testing each model on the opposing group's data, accuracy fell considerably, suggesting that physically active and inactive populations exhibit sufficiently different patterns to challenge a model trained on the other group.

Overall, the results paint a picture of moderate success and reveal important considerations for transferring models across groups that differ in relevant lifestyle factors. While the SVM approach has proven robust within individual domains, practitioners should exercise caution before applying a single model across populations with distinct behaviors or traits.

In future work, exploring a wider range of hyperparameters, incorporating additional features, or investigating domain-adaptation techniques could help augment generalizability. Nevertheless, this investigation underscores how powerful machine learning can be for timely diabetes detection when aligned appropriately with the characteristics of the target population.