SVM Vs Logistic Regression Vs kNN

Rohan Sikder – G00389052

Step 1: Initial Data Exploration
To get started, Firstly load and take a look at the dataset to understand its structure and types of data it contains.

Dataset
The dataset includes driving data from different vehicles with 23,775 observations. Each observation represents a set of vehicle and driving characteristics at a specific moment in time, gathered from vehicles such as Opel Corsa and Peugeot 207 during driving sessions. The dataset contains 15 input variables (features) representing different aspects of vehicle dynamics, engine parameters and environmental conditions. Predicted variable drivingStyle is a measure of driving behaviour that divides driving behaviour into two classes EvenPaceStyle, and AggressiveStyle. This classification could allow for modelling of driving behaviours which may be needed for applications such as insurance or fuel efficiency.

Step 2: Data Cleaning

- **Eliminate Redundant Columns:** Columns like Unnamed: 0, which is likely part of the data export process will be removed for accuracy and efficiency.
- **Adjust Data Types:** Some columns expected to contain numbers may still be treated as strings by using formatting details such as decimal commas. These will be correctly transformed to floating point numbers for easier analysis.
- **Manage Missing Data:** Whether to fill in these missing values with The median or mean of the columns in question or to leave out. In this case the missing data rows will be dropped.

Step 3: Data Pre-processing

- **Feature Scaling:** Apply **StandardScaler** to normalize the feature set. This step is crucial for models like SVM and kNN which are sensitive to the scale of input features, ensuring all features contribute equally to model training.
- **Splitting the Dataset:** Divide the dataset into training and test sets with 80% of the data allocated for training and the remaining 20% for testing. This split is essential for evaluating the model's performance on unseen data.

Step 4: Model Training and Evaluation
Train and compare three different classifiers: SVM, Logistic Regression, and kNN. For each model:

- **SVM:** Utilizing the SVM classifier with a linear, rbf, ploy and sigmoid kernel.
- **Logistic Regression:** Applying logistic regression with class weight balanced to address any class imbalance.
- **k-Nearest Neighbors (kNN):** Implementing the kNN classifier starting with a default number of neighbours.

Cross-validation is a technique for assessing machine learning models performance. Its main objective is making sure that the model generalizes well to unknown data, reducing the overfitting risk. Overfitting occurs when a model learns the training data too closely capturing noise together with the underlying pattern and then failing to perform well on new data.

K-fold cross-validation is a common kind of cross-validation. For k-fold cross-validation, the data set is split into k equal subsets or folds. The model is tested on k-1 folds and then on the remaining fold. This is repeated k times and each fold is the test set once. The model performance is then averaged over these k trials to give a more complete picture of its effectiveness and reliability.

Random Forest is a method that combines many decision trees to find out which parts of the data are most important for figuring out driving style. Random Forest helps to see which features like vehicle speed or how the engine runs matter most by looking at how much each feature helps improve the predictions across all the trees. This step was not about comparing it directly with SVM, Logistic Regression, and kNN in terms of accuracy. Instead, it helped to get a clearer picture of which car behaviours are key indicators of driving style making it easier to understand how the main models classify driving styles based on these behaviours.

Accuracy Comparison:

- The kNN model achieved the highest overall accuracy of 94%.
- SVM with RBF kernel closely followed with an accuracy of 80%.
- Logistic Regression model had the lowest accuracy at 69%.

Analysis of Precision, Recall and F1-Score:

- kNN demonstrated the highest precision for AggressiveStyle (75%) and EvenPaceStyle (96%), showing its ability to accurately classify instances of both classes.
- SVM with RBF kernel obtained the best recall for AggressiveStyle (88%) and EvenPaceStyle (79%), showing its efficiency in capturing a large proportion of real instances of both classes.
- The kNN model obtained the best F1-score of both classes, showing a good balance between precision and recall.
- Logistic Regression returned the lowest precision, recall and F1-score of both classes for both classes showing that it was having difficulty in correctly identifying instances of both classes.

Model Specifics and Feature Importance:

- Logistic Regression was also outperformed by SVM with RBF kernel and kNN models, indicating that non-linear decision boundaries are probably better suited for this classification task.
- Random Forest model feature importance analysis indicated which features were more influential in driving model performance pointing to potential insight into driving behaviour.

Key Takeaways:

- The kNN model offered the best balance between accuracy, precision, recall, and F1-score.
- SVM with RBF kernel performed well, especially in capturing actual instances of both classes.
- Logistic Regression showed the lowest performance among the three models, indicating potential limitations in its ability to handle the complexity of the dataset.

Conclusion

Finally comparing of SVM, Logistic Regression and kNN models for classifying driving styles. Results showed that kNN model had the best overall accuracy of 94% better than SVM and Logistic Regression models. Despite its simplicity kNN showed better precision and recall for aggressive and even-pace driving styles suggesting that it was capable of correctly classifying instances of each type.

SVM with RBF kernel closely followed kNN in accuracy to achieve 80% accuracy. This model showed good recall for both driving styles especially aggressive driving. However, it was less accurate than the kNN model.

The computationally efficient Logistic Regression performed the worst of the three models with an accuracy of 69%. Its lower precision, recall and F1-score indicate limitations in handling the dataset complexity particularly in capturing examples of both driving styles with sufficient precision.

The kNN model obtained the balance between accuracy, precision, recall and F1-score and is thus suitable for the classification of driving styles in the dataset.

References

1. Support Vector Machine (SVM):
    - **Title:** What Is a Support Vector Machine? Working, Types, and Examples
    - **Source:** Spiceworks
    - **URL:** https://www.spiceworks.com/tech/big-data/articles/what-is-support-vector-machine/
2. K-Fold Cross-Validation Technique:
    - **Title:** K-Fold Cross Validation Technique and its Essentials
    - **Source:** Analytics Vidhya
    - **URL:** https://www.analyticsvidhya.com/blog/2022/02/k-fold-cross-validation-technique-and-its-essentials/
3. Using **StandardScaler()** Function:
    - **Title:** Using StandardScaler() Function to Standardize Python Data
    - **Source:** DigitalOcean
    - **URL:** https://www.digitalocean.com/community/tutorials/standardscaler-function-in-python
4. Understand Random Forest Algorithms:
    - **Title:** Understand Random Forest Algorithms
    - **Source:** Analytics Vidhya
    - **URL:** https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/