Category Classification & Missing Value Estimation Project Report

Maja Kadic

Department of Computer Science CSC 4850

Georgia State University

[mkadic2@student.gsu.edu](mailto:mkadic2@student.gsu.edu)

# Introduction

In the evolving landscape of data science and analytics, the ability to accurately classify and estimate missing values in complex datasets is of paramount importance. This project report delves into these critical aspects, using seven distinct datasets as the basis for exploration and analysis.

# Approach and Methodology

Our project is centered around two key tasks: classification of complex datasets and missing value estimation. We employ distinct methodologies for each of the seven datasets under study, tailored to their unique characteristics in terms of dimensionality, sample size, and class distribution.

X

# Classification

For each dataset, a primary classification method has been chosen based on specific dataset characteristics, along with an alternative method for comparison.

1. **Dataset 1 (High Dimensionality, Small Sample Size, 5 Classes):**
   * **Primary Method**: Principal Component Analysis[4] + Support Vector Machine[5]
   * **Predicted Impact**: Principal Component Analysis will reduce dimensionality, mitigating the risk of overfitting due to the small sample size. Support Vector Machine is effective for classification and can handle the non-linear relationships possibly present in high-dimensional data.
   * **Rationale**: Principal Component Analysis mitigates the risk of overfitting in smaller datasets by reducing dimensionality, whereas Support Vector Machine adeptly manages non-linear relationships within the data. PCA combined with SVM tends to be better suited for smaller datasets, as it typically offers a more generalized approach and helps in avoiding overfitting.
   * **Alternative Method**: Random Forest[3] - can also handle high dimensionality but might be less effective with a small sample size compared to the more focused Support Vector Machine approach.
   * **Results**:
2. **Dataset 2 (Very High Dimensionality, Moderate Sample Size, 11 Classes):**
   * **Primary Method**: Principal Component Analysis + Gradient Boosting[1]
   * **Predicted Impact**: Principal Component Analysis reduces dimensionality, essential for very high-dimensional data. Gradient Boosting is a strong classifier, especially effective for datasets with many classes.
   * **Rationale**: Principal Component Analysis is critical for dimensionality reduction, and Gradient Boosting excels in multi-class scenarios. Both together, offer a more robust approach for moderate sample sizes.
   * **Alternative Method**: Neural Network - requires larger datasets to perform optimally and might overfit or require extensive tuning in this scenario.
   * **Results**:
3. **Dataset 3 (Low Dimensionality, Large Sample Size, 9 Classes):**
   * **Primary Method**: k-NN[2]
   * **Predicted Impact**: k-NN works well with large datasets and lower dimensionality, providing high accuracy through its simplicity.
   * **Rationale**: k-NN's simplicity and effectiveness in classifying large, less complex datasets make it a suitable primary choice.
   * **Alternative Method**: Random Forest - potentially excessive for low complexity data and could lead to longer training times without significant accuracy gains.
   * **Results**:
4. **Dataset 4 (Moderate Dimensionality and Sample Size, 9 Classes):**
   * **Primary Method**: Random Forest
   * **Predicted Impact**: Offers a good balance of bias-variance, handles moderate dimensionality well, and is robust to overfitting.
   * **Rationale**: Balances bias-variance and is robust against overfitting. Random Forest is a versatile, less complex, and faster model, suitable for moderate-sized datasets.
   * **Alternative Method**: Gradient Boosting - could potentially offer similar results but might require more fine-tuning and is generally slower to train.
   * **Results**:
5. **Dataset 5 (Low Dimensionality, Moderate Sample Size, 6 Classes):**
   * **Primary Method**: k-NN
   * **Predicted Impact**: k-NN is effective in scenarios with lower dimensionality and can perform well with a moderate sample size.
   * **Rationale**: Effectivity in handling multi-class problems in lower dimensional spaces, along with its simplicity make it a preferable choice in this context.
   * **Alternative Method**: Logistic Regression with regularization - also a strong classifier but may not perform as well as k-NN in capturing complex relationships in multi-class scenarios.

X

* + **Results**:

# Missing Value Estimation

Accurate estimation of missing values is critical to ensure the integrity of our analysis.

1. **Dataset 1**: We utilize Mean/Median Imputation[6], aiming to fill 4% of missing values efficiently without distorting the data distribution.
   * **Results**:
2. **Dataset 2**: k-NN Imputation is deployed to estimate 10% missing values, leveraging sample similarity for greater accuracy.

X

* + **Results**

1. **Dataset 3:**

# Results

The chosen methodologies are expected to optimize the classification accuracy while maintaining the integrity of the datasets. The combination of Principal Component Analysis with Support Vector Machine or Gradient Boosting is anticipated to effectively manage high dimensionality challenges. k-NN's simplicity is projected to yield high accuracy in datasets with large sample sizes or lower complexity. Random Forest, employed in moderate scenarios, should provide a balanced approach. The missing value estimation strategies are designed to maintain the datasets' original structure and distribution, ensuring reliable and robust classification results.

X

# Conclusion

This approach and methodology section lays out a structured and strategic plan to tackle the complexities inherent in the datasets. By carefully selecting and applying these methods, the project aims to yield accurate classification and missing value estimation results.

X

# References

[1] “Gradient Boosting Machines,” Gradient Boosting Machines · UC Business Analytics R Programming Guide, https://uc-r.github.io/gbm\_regression (accessed Nov. 13, 2023).

[2] T. M. Mitchell, “k-NEAREST NEIGHBOR LEARNING,” in *Machine learning*, McGraw-Hill, 1997, pp. 231–236

[3] “Random Forest: A complete guide for machine learning,” Built In, https://builtin.com/data-science/random-forest-algorithm (accessed Nov. 13, 2023).

[4] J. Brownlee, “How to calculate principal component analysis (PCA) from scratch in Python,” MachineLearningMastery.com, https://machinelearningmastery.com/calculate-principal-component-analysis-scratch-python/ (accessed Nov. 13, 2023).

[5] “1.4. Support Vector Machines,” scikit, https://scikit-learn.org/stable/modules/svm.html (accessed Nov. 13, 2023).

[6] Z. Zhang, “Missing data imputation: Focusing on single imputation,” Annals of translational medicine, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4716933/ (accessed Nov. 13, 2023).