CS F320: Foundations of Data Science
Project
Analysis and Insights

Vidit Benjwal (2021A7PS0004P)
Akshit Phophaliya (2021B3A71738P)
Anirudh Anand (2021B3A70981P)
Vishnu Hari (2022A7TS0094P)

November 21, 2024

Introduction

- Objective: Develop and analyze regression, classification, and clustering models to predict blood glucose levels and diabetes risk
- Focus: Model performance, feature engineering, and result interpretation.

Libraries and Tools

- **pandas**: For data manipulation and analysis.
- **requests**: To download datasets.
- **xgboost, ngboost, adaboost**: Gradient boosting.
- **scikit-learn-extra**: Additional machine learning utilities.
- **scipy.spatial.distance**: For distance calculations.

Dataset Loading and Preprocessing

- Dataset downloaded from Google Drive using the `requests` library and loaded into a `pandas` DataFrame.
- Data is then preprocessed by using Principal Component Analysis (PCA) to identify the top 3 principal components
- Results shown below indicate
  - PC1: Captures the most variance from the original dataset. Values indicate that the observation strongly aligns with this direction.
  - PC2: Represents the second most important direction of variance. Some values show notable contributions.
  - PC3: Captures less variance than PC1 and PC2. Some entries suggest the observation is negatively correlated with this principal component.

Techniques and Results

- Regression
  - Linear Regression
  - Random Forests and Extra Random Trees
  - Gradient Boosting Techniques: XGBoost, NGBoost
  - Adaptive Boosting Techniques: AdaBoost

Model	RMSE	MAE
Linear Regression	47.355947	36.558859
Random Forest	48.240764	37.331256
Extra Trees	48.083156	36.698548
AdaBoost	47.910267	38.676403
XGBoost	52.169931	40.474213
NGBoost	47.434920	35.557379

Inference: Regression

- From the given results, we see the following observations:
  - Linear Regression and NGBoost show competitive performance with respect to RMSE and MAE, respectively. This suggests that NGBoost might perform better when focusing on reducing absolute errors, while Linear Regression is more effective at reducing squared errors.
  - XGBoost has the highest RMSE (52.169931) and MAE (40.474213), indicating it performs the worst among the listed models on this dataset.
  - Linear Regression is better suited for minimizing large errors (emphasized by RMSE).
  - NGBoost is better suited for minimizing average errors (emphasized by MAE).

Techniques and Results

- Classification
  - Support Vector Machines (SVMs): Linear SVM, Kernel SVM
  - Logistic Regression
  - k-Nearest Neighbours (kNN)
  - Decision Trees
  - Basic Neural Networks

Model	Precision (Original)	Recall (Original)	F1-score (Original)	Precision (PCA)	Recall (PCA)	F1-score (PCA)
Logistic Regression	0.866197	0.736527	0.796117	0.856164	0.748503	0.798722
Naive Bayes	0.882812	0.676647	0.766102	0.883333	0.634731	0.738676
K-Nearest Neighbors	0.796992	0.634731	0.706667	0.861842	0.784431	0.821317
Linear SVM	0.882353	0.718563	0.792079	0.876812	0.724551	0.793443
Kernel SVM	0.907895	0.413174	0.567901	0.909091	0.718563	0.802676
Decision Trees	0.838710	0.778443	0.807453	0.774194	0.718563	0.745342
Neural Networks	0.920000	0.688623	0.787671	0.888889	0.766467	0.823151

Inference: Classification

- From the given results, we see the following observations:
  - PCA improved performance for K-Nearest Neighbors, Kernel SVM, and Neural Networks, making them strong candidates when dimensionality reduction is applied.
  - Kernel SVM and Neural Networks saw the most substantial improvement in Recall and F1-score after PCA.
  - PCA had a negative impact on Naive Bayes and Decision Trees, with reduced F1-scores, suggesting these models are less suited for datasets with PCA transformation.
  - Logistic Regression and Linear SVM maintained stable performance across both datasets, indicating robustness to dimensionality changes.
  - For high-precision tasks, Neural Networks or Kernel SVM are top picks. If recall is crucial, K-Nearest Neighbors (PCA) excels. For general performance, Neural Networks (PCA) or Decision Trees (Original) are strong choices.

Techniques and Results

- Z-Score Transformation: Standardizes data by scaling features to have a mean of 0 and a standard deviation of 1. It ensures that features with different units or scales contribute equally in our clustering models.
- Clustering Techniques
  - k-Means (k=2)
  - k-Medoids
  - EM Clustering

Clustering Method	Silhouette Score	Dunn's Index	BetaCV	Hubert Statistic	SSE
K-means (Original)	0.563312	0.308604	2.283016	0.264006	3569228.444712
EM (Original)	0.387285	0.001843	1.517056	0.366783	N/A
K-medoids (Original)	0.570413	0.018577	2.493087	0.321107	N/A
K-means (Z-score)	0.256037	0.014794	1.372868	0.368408	7096.868923
EM (Z-score)	0.250315	0.009912	1.294957	0.394090	N/A
K-medoids (Z-score)	0.277701	0.009980	1.381920	0.572792	N/A

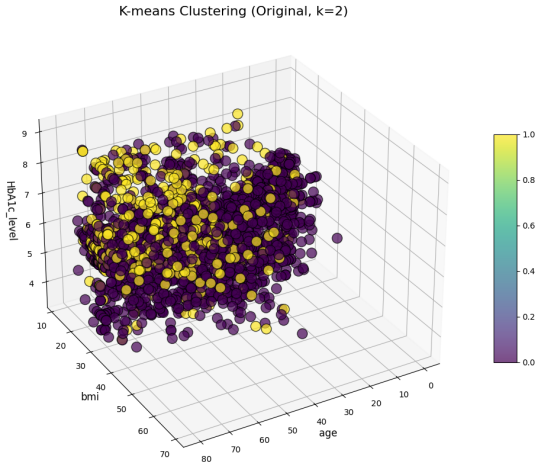
Statistics Calculated

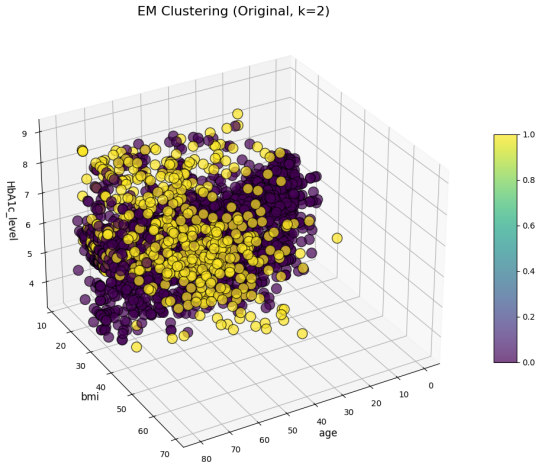
- SSE (Sum of Squared Errors)
- Silhouette Score: Quality of Clustering, higher values mean better quality clustering
- BetaCV (Beta Cross-Validation): Stability of Clustering, higher values mean less stable clusters and higher intra-cluster distance and/or lower inter-cluster distance
- Dunn's Index: Compactness of clusters and the separation between them, higher values indicate better-defined, more distinct clusters.
- Hubert's Statistic: Evaluates the agreement between two clustering results, with values closer to 1 indicating high agreement between different clusterings.

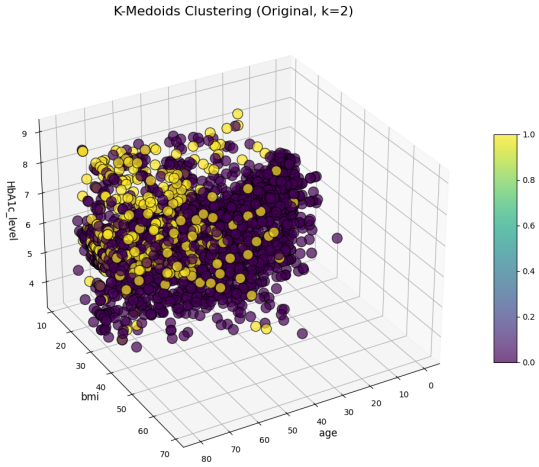
Inference: Clustering

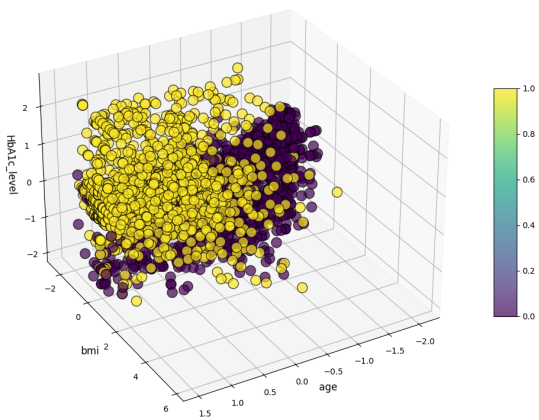
- From the given results, we see the following observations:
  - BetaCV and Silhouette Scores decrease by 2-fold after Z-score normalization, likely due to altered distance metrics affecting cluster structures.
  - A significant reduction in SSE (500x) is observed for normalized data in k-means, showing the benefit of Z-score normalization scaling in distance-sensitive techniques.
  - But K-means performs poorly in both datasets, because it is hindered by sensitivity to the initial centroid and inability to manage non-globular clusters.
  - Hubert Statistic is chosen as the most reliable metric. This is because due to its invariance to scaling, it measures cluster alignment with true labels effectively.
  - EM clustering excels in capturing data patterns due to its probabilistic approach, making it the best performer without normalization.
  - K-Medoids outperforms all other methods after Z-Score normalization, showcasing its robustness to scaling and ability to handle diverse data distributions.

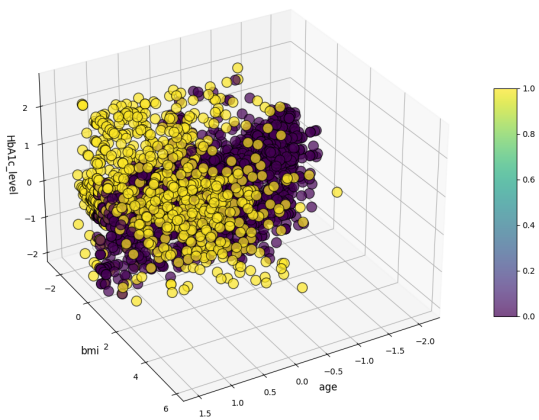
Visual Representation: Clustering Results

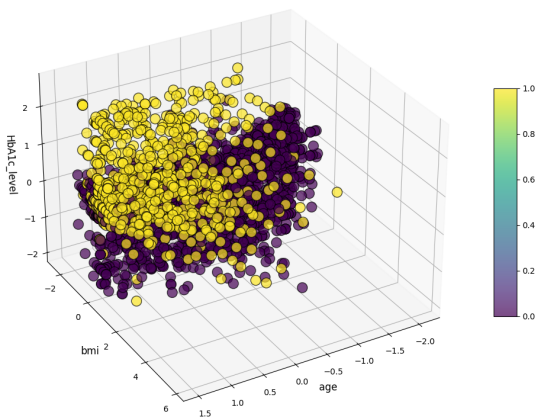












Thank You!

Any questions?