# Report: Video Game Sales Prediction

## Cover Page

Title: Video Game Sales Prediction Report
Author: [Mohammed Ashraf Mohammed]
Date: [28/12/2023]

## Definition of the Problem

The video game industry is dynamic and highly competitive. Predicting the sales performance of video games is crucial for game developers, publishers, and stakeholders. This report aims to explore and implement regression models to predict global video game sales based on various features such as 'Name', 'Platform', 'Year', 'Genre', 'Publisher', 'NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales', and 'Global_Sales'.

## Method

### Data Collection

We collected a dataset containing information on video game sales, including the aforementioned features. The dataset provides a comprehensive overview of the industry, allowing us to build predictive models.

### Data Preprocessing

Prior to model development, we conducted data preprocessing steps like data visualization and data cleaning, including handling missing values, encoding categorical variables, and scaling numerical features.

## Experiment

We employed several regression models to predict global video game sales:

Linear Regression Model

Decision Trees

Random Forest Regressor

Support Vector Regressor (SVR)

Gradient Boosting Regressor

MLPRegressor (Multi-layer Perceptron Regressor)

For each model, we evaluated performance metrics such as Mean Squared Error (MSE), R-squared, and model interpretability.

References

*1:https://www.kaggle.com/datasets*

*2:*https://scikit-learn.org/

# Report: Employee Retention Prediction

## Cover Page

Title: Employee Retention Prediction Report
Author: [Mohammed Ashraf Mohammed ]
Date: [28/12/2024]

## Definition of the Problem

Employee retention is a critical aspect of organizational success. This report focuses on predicting employee retention based on various factors such as 'Education', 'JoiningYear', 'City', 'PaymentTier', 'Age', 'Gender', 'EverBenched', 'ExperienceInCurrentDomain', 'LeaveOrNot', 'Gender_encoded', 'EverBenched_encoded', and 'Education_encoded'. The objective is to build effective classification models that can identify potential instances of employee turnover.

## Method

### Data Collection

We collected a dataset containing information about employees, including the specified features. The dataset aims to capture relevant factors influencing employee retention.

### Data Preprocessing

Preprocessing steps involved handling missing values, encoding categorical variables, and ensuring numerical features are appropriately scaled.

## Experiment

We applied various classification models to predict employee retention:

Logistic Regression

Decision Tree Classifier

Random Forest Classifier

Support Vector Classifier (SVC)

K-Nearest Neighbors Classifier (KNeighbors)

Multi-layer Perceptron Classifier (MLPClassifier)

For each model, we assessed performance metrics such as accuracy, precision, recall, and F1-score. Additionally, we considered model interpretability and potential business implications.

# References

1:https://www.kaggle.com/datasets

2: https://scikit-learn.org/

# Report: Exploring Diabetes Dataset with Dimensionality Reduction

## Cover Page

Title: Diabetes Dataset Exploration with Dimensionality Reduction
Author: [Mohammed Ashraf Mohammed ]
Date: [Date]

## Definition of the Problem

This report aims to explore a diabetes dataset comprising the features 'Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age', and 'Outcome'. The primary goal is to utilize dimensionality reduction techniques to gain insights into the underlying patterns and structures within the data.

## Method

### Data Collection

We gathered a dataset containing information on diabetes patients, including the specified features. The dataset provides valuable information for understanding the relationships between different health metrics and the presence or absence of diabetes.

### Data Preprocessing

Preprocessing steps included handling missing values, standardizing features, and ensuring the dataset is suitable for dimensionality reduction techniques.

## Experiment

We applied the following dimensionality reduction techniques:

Principal Component Analysis (PCA)

Independent Component Analysis (ICA)

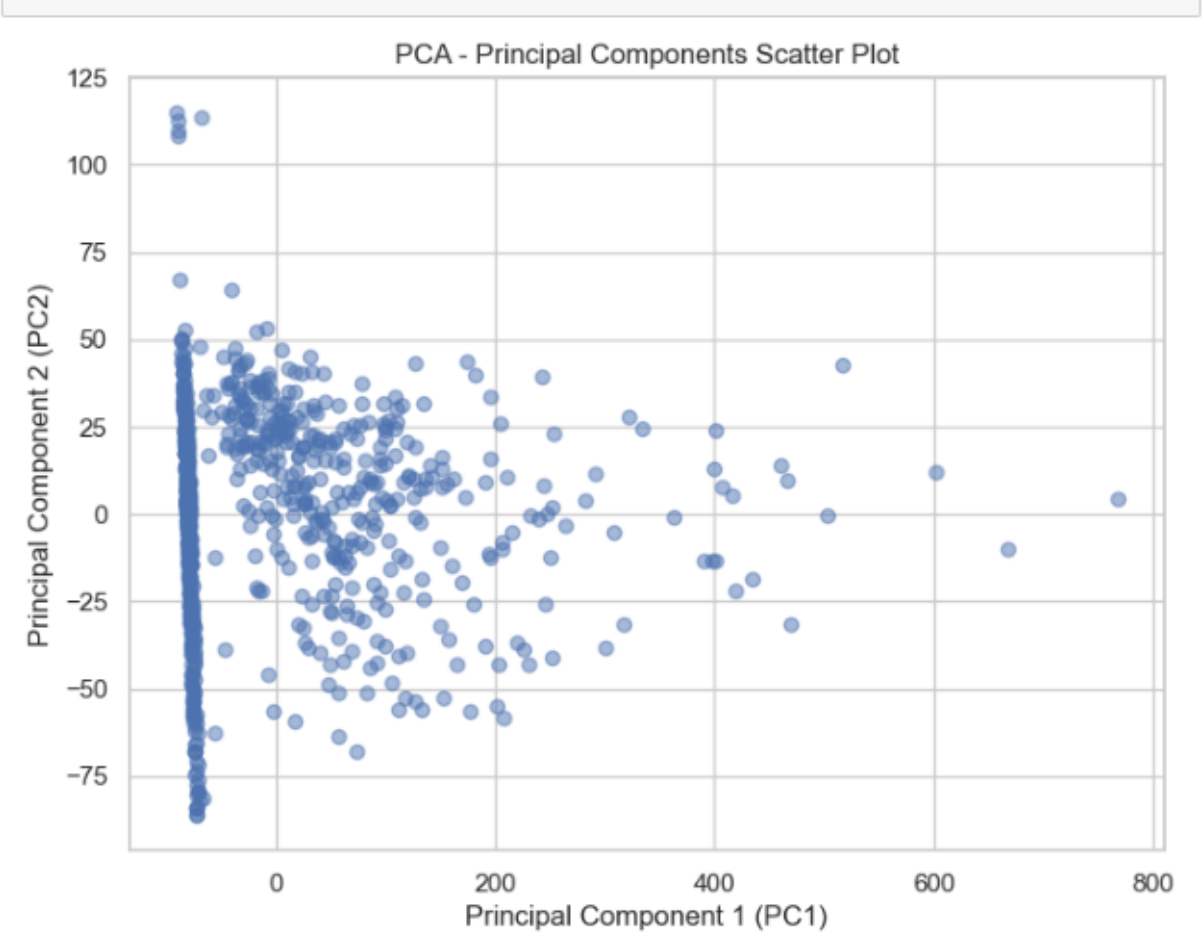
DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

t-Distributed Stochastic Neighbor Embedding (t-SNE)

Each technique was employed to reduce the dimensionality of the dataset, and the results were analyzed to uncover patterns and relationships in the data.

## Results

# Principal Component Analysis (PCA)




**PCA Insights and Findings**

The principal components offer a compressed representation of the original data, highlighting the directions of maximum variance. Key insights from the PCA analysis include:

PC1 Impact:

The negative values in PC1 indicate a consistent decrease in the values of the original features for observations like 0, 1, and 2.

Positive values in PC1 (e.g., observations like 4 and 763) suggest an increase in the original feature values.

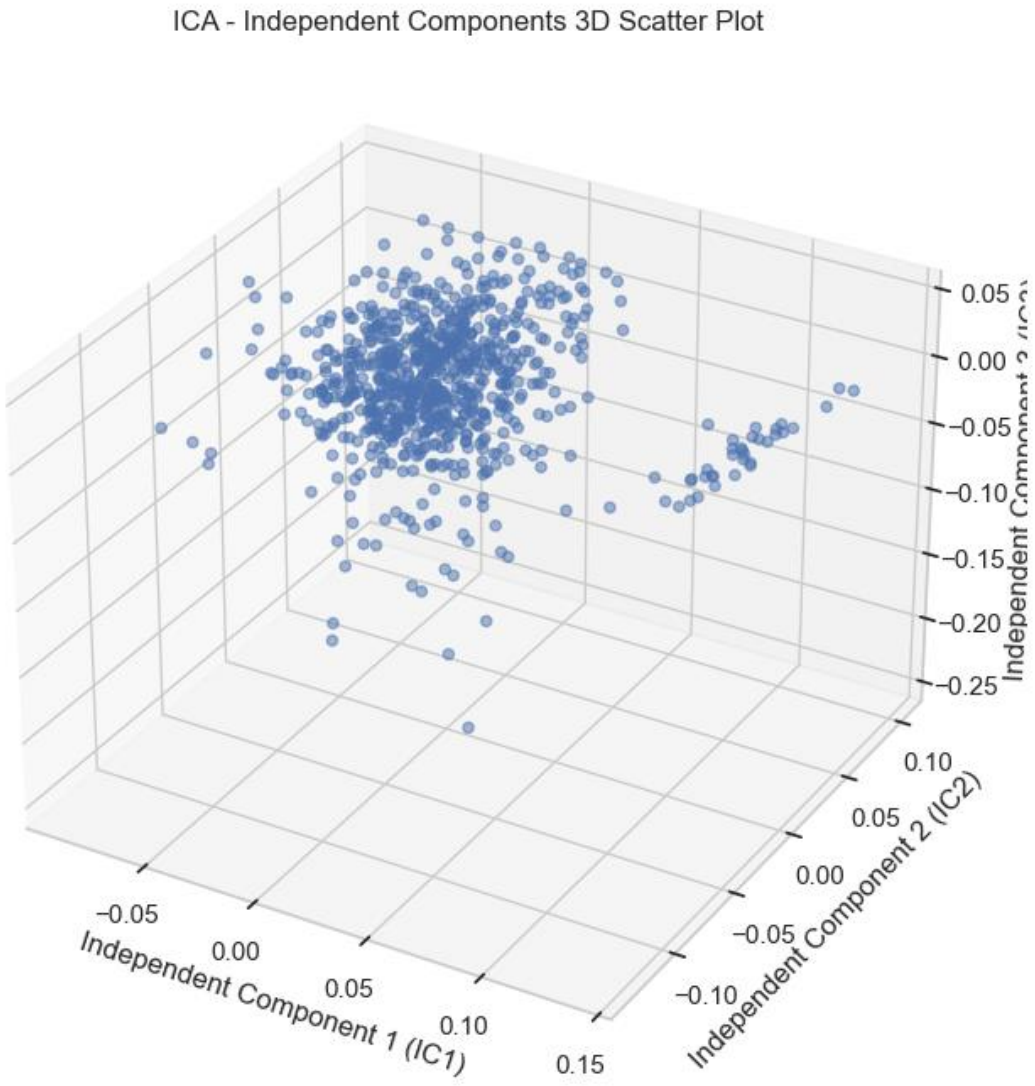
PC2 Influence:

Positive values in PC2 (e.g., observations like 1, 3, and 767) indicate an increase in different features compared to other observations.

Negative values in PC2 (e.g., observations like 2, 4, and 766) suggest a decrease in different features.

These findings provide a valuable understanding of how observations are distributed in the reduced feature space defined by PC1 and PC2. Further analysis, clustering, or visualization can be conducted to explore patterns and relationships in the data.

## Independent Component Analysis (ICA)



ICA Insights and Findings

The independent components derived from ICA offer a unique representation of the original features. Key insights from the ICA analysis include:

IC1 Impact:

IC1 appears to capture variations related to a combination of features, with both positive and negative values.

IC2 Influence:

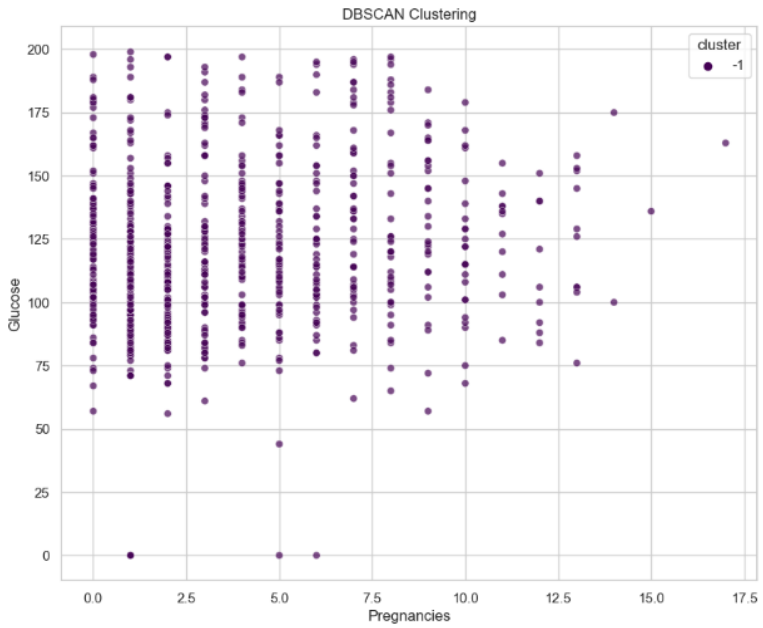
IC2's positive and negative values suggest variations in different features, contributing to the overall independent structure.

IC3 Patterns:

IC3 indicates specific patterns that might represent variations in the original features.

These findings provide a valuable understanding of how independent components contribute to the data's structure. Further analysis, clustering, or visualization can be conducted to explore patterns and relationships in the data based on these independent components.

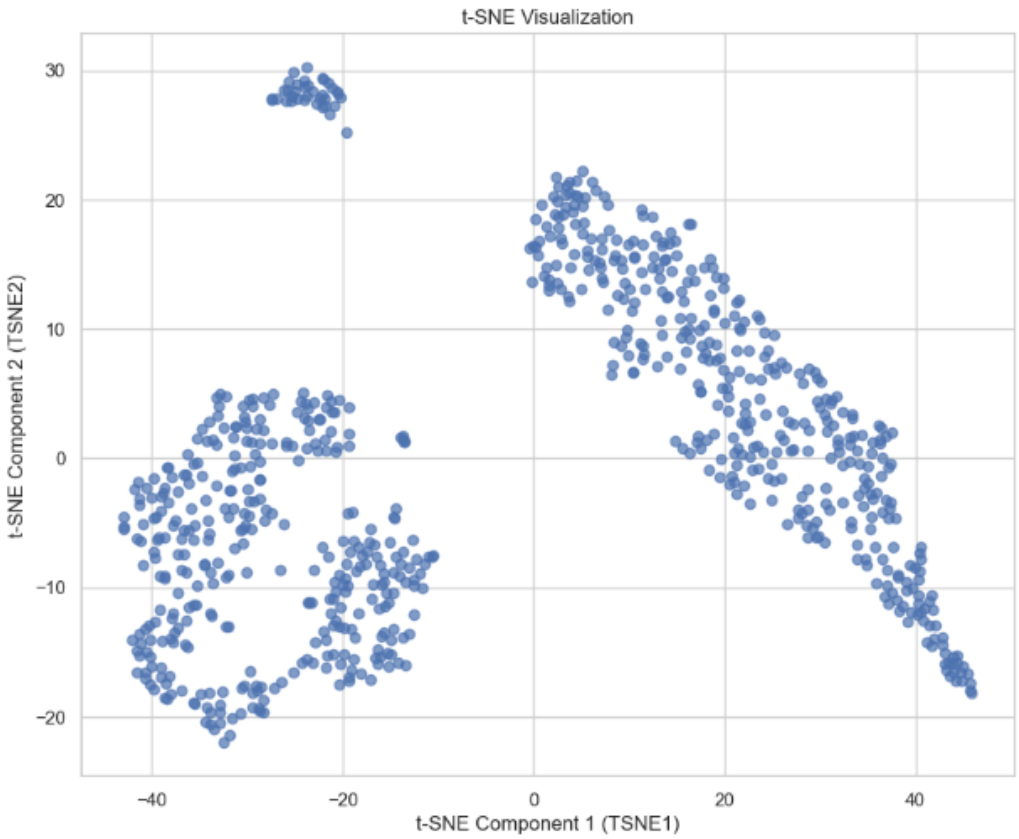
## DBSCAN



**DBSCAN analysis:**

The DBSCAN algorithm assigned most data points to the cluster label -1, indicating noise or outlier points. This suggests that the majority of the data does not conform to dense regions, and the algorithm identified these points as not belonging to any specific cluster.

## t-Distributed Stochastic Neighbor Embedding (t-SNE)




t-SNE Insights and Findings:

t-SNE provides a non-linear mapping of high-dimensional data to a lower-dimensional space, allowing us to visualize complex relationships. Key insights from the t-SNE analysis include:

Distribution of Data:

Observations with similar characteristics tend to cluster together in the t-SNE space.

Local Structure Preservation:

The proximity of points in the t-SNE space may reflect the local structure and relationships present in the original high-dimensional data.

**Visualization of Clusters:**

t-SNE visualization can aid in identifying potential clusters or groups within the dataset.

These findings offer a valuable perspective on the intrinsic structure of the data in a reduced-dimensional space.

# References

1:https://www.kaggle.com/datasets

2: https://scholar.google.com/

*Cover Page:*

*Title: "Clothes Classification using Convolutional Neural Networks (CNN)"*

*Mohammed Ashraf Mohammed*

*1. Definition of the Problem:*

*1.1 Introduction:*

*Automated clothes classification plays a crucial role in various applications, offering efficiency and accuracy over manual methods. This project focuses on implementing a Convolutional Neural Network (CNN) to classify clothing items.*

*1.2 Problem Statement:*

*The manual classification of clothes is time-consuming and prone to errors. Automating this process using CNNs can enhance accuracy and speed, benefiting applications such as retail, inventory management, and fashion analysis.*

*1.3 Objectives:*

*Implement a CNN model for clothes classification.*

*Evaluate and compare the model's performance with predefined metrics.*

*Showcase the effectiveness of automated clothes classification in real-world scenarios.*

*2. Method:*

*2.1 Overview of Convolutional Neural Networks (CNNs):*

*CNNs are a type of neural network particularly suited for image classification tasks. The chosen architecture involves convolutional layers for feature extraction and pooling layers for down-sampling.*

***2.2 Data Exploration:*** The dataset consists of 60,000 entries with 785 columns, where each entry represents a clothing item. The labels are stored in the 'label' column, and the pixel values are spread across 784 columns. The training set comprises 48,000 entries, and the test set comprises 12,000 entries.

***2.3 Data Collection:***The dataset comprises 48,000 training samples and 12,000 test samples, each with a dimension of (28, 28) representing grayscale images of clothes. Preprocessing steps include resizing and normalization.

**2.4 Data Visualization:** A random selection of 25 clothing items from the training set is visualized using matplotlib. The images are displayed along with their corresponding labels. The class names include 'T-shirt/top,' 'Trouser,' 'Pullover,' 'Dress,' 'Coat,' 'Sandal,' 'Shirt,' 'Sneaker,' 'Bag,' and 'Ankle boot.'

**2.5 Data Preprocessing:** The images in the dataset are reshaped to (28, 28) and normalized, resulting in a training set of shape (48000, 28, 28) and a test set of shape (12000, 28, 28). The pixel values are converted to floating-point numbers between 0 and 1.

**2.6 Model Architecture:**

1st Layer: Conv2D with 64 filters, kernel size 3, valid padding, and ReLU activation, followed by MaxPooling2D.

2nd Layer: Conv2D with 128 filters, kernel size 3, same padding, and ReLU activation, followed by MaxPooling2D.

3rd Layer: Conv2D with 64 filters, kernel size 3, same padding, and ReLU activation, followed by MaxPooling2D.

Flattening layer followed by Dense layers with dropout for regularization.

Output layer with 10 units and softmax activation.

2.7 Training:

Training parameters include learning rate, batch size, and epochs. Data augmentation techniques are applied to enhance model generalization.

3. Experiment:

*The implemented CNN model demonstrates significant success in classifying clothing items, achieving an overall accuracy of 90.22% on the test set. The precision, recall, and F1-score metrics provide detailed insights into the model's performance for each class. The model can be further optimized or fine-tuned based on the specific requirements of the application.*

4. References:

*1:* https://scikit-learn.org/stable/documentation.html

2: https://www.tensorflow.org/

*3:* https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-fashion-mnist-clothing-classification/