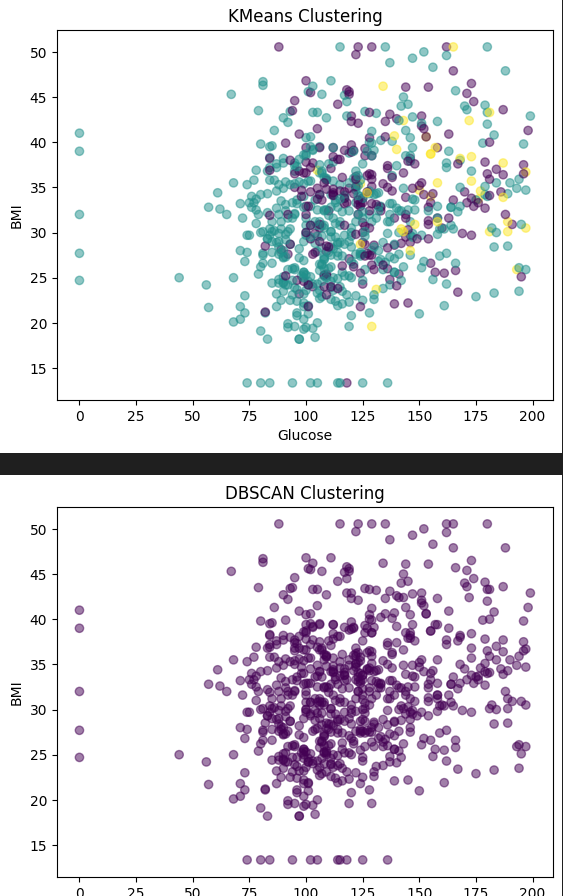
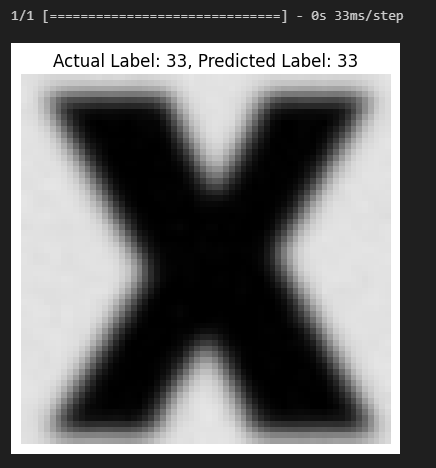
**Project Documentation: Machine Learning Projects**  
Project 1: Supervised Learning Regression (Structured Dataset)  
Loading and Exploring Data:  
We started by reading the required libraries such as Pandas, NumPy, Seaborn and Matplotlib , in addition to machine learning modules from Scikit-learn . We imported and analyzed the ‘housing\_data.csv’ with Pandas, reviewed first rows, dataset information metadata statistics summary tabl。 There were visualizations, such as pair plots, correlation matrices, box plots and histograms that enabled the understanding of numerical column relationships and distributions.  
  
Data Preprocessing:  
We used one-hot encoding of categorical variables through ‘get\_dummies’ and selected numerical columns with Scikit-learn’s MinMaxScaler. We imputed the mean to handle missing values and performed Principal Component Analysis (PCA) on these columns.  
  
Cleaning Data and Feature Engineering:  
We reformatted the ‘date’ column to a datetime format, made sure of consistency between ‘code’ and ’ area' columns by rectifying them where necessary. We also pulled out fresh features such as ‘year’ and ‘month” from the column marked “date”.  
  
Training and Evaluating Models:  
We divided the data in features X and target variable y, based on that we applied different regression models for predicting ‘no\_of\_crimes’; performance measures like Mean Squared Error (MSE) & R ². Always used training test splits.  
  
Improving Our Models:  
We fine-tuned a RandomForestRegressor model with optimal parameters based on the previous assessments and employed it for prediction of ‘no\_of\_crimes’ variable, including these predictions in the housing\_data DataFrame.  
  
Hyperparameter Tuning:  
We used GridSearchCV to find the best hyperparameters for RandomForestRegressor model.  
  
Final Model Evaluation:  
The RandomForestRegressor model with tuned parameters was tested on test data set. Performance of the mode is measured by means MSE and R-squared criteria.

Project 2: Supervised Learning Classification (Structured Dataset)  
Preparing Tools:  
We used tools for data handling and model generation.  
  
Checking Our Data:  
We loaded a gender classification dataset, looked through its contents to view missing information and found duplicates which were removed.  
  
Getting Our Data Ready:  
We adapted information types to improve usability and changed words or labels into numerical formats for better understanding by machines.  
  
Understanding Our Data:  
We plotted relationships in the data using graphs, analyzing which characteristics were widespread.  
  
Building and Testing Models:  
Data was divided to train different models in which they were able to predict gender using the provided information. The gender predictions of each model were tested to see how true they are.  
  
Summarizing Our Approach:  
We have considered various approaches to mean the sense and determinate of gender, evaluated them taken into prediction account on this particular purpose.  
  
Project 3: Unsupervised Learning (Structured Dataset)  
Loading and Exploring Data:  
We imported Pandas, loaded the diabetes dataset ‘patients.csv’, and first checked its structure summary statistics as well as the missing values within it.  
  
Data Inspection and Visualization:  
Further investigation included missing values, summary statistics and data types along with visualizations such as count plots or heat maps for the understanding of distributions and relationships.  
  
Feature Analysis and Visualizations:  
We went for a more in-depth exploration of crucial characteristics, including ‘Glucose’,’ BMI ’,’ Age ’ ,‘ Insulin – and ‘BloodPressure; using box plots as well as pair plots.  
  
Data Preprocessing:  
The handling of outliers, particularly for the ‘BMI’ column, was done through replacing values that went beyond specific limits.  
  
Clustering Techniques:  
To identify patterns within the dataset, K-Means and DBSCAN clustering methods were employed; these allowed clusters to be visualized based on ‘Glucose’ and ‘BMI’.  
  
Dimensionality Reduction:  
Using t-SNE , we managed to minimize dataset dimensions for easy visualization.  
  
Further Exploration:  
We explored the unique values contained in ‘Age’ and ‘Pregnancies’ columns more deeply.  
  
Independent Component Analysis (ICA):  
Using FastICA, we extracted original features into statistically independent components in order to discover underlying factors.

Project 4: 1. Supervised Learning Classification  
Data Loading and Preprocessing:  
We loaded images with labels using OpenCV and NumPy, resized them, converted to float format 0-1 scale.  
  
Data Preparation:  
Images were converted into NumPy array and labels encoded using LabelEncoder(). The dataset was shuffled to make it random.  
  
Building the Convolutional Neural Network (CNN):  
An image classification model with sequential convolutional layers, max-pooling layers and dense was built using Keras.  
  
Compiling and Training the Model:  
The model was compiled by using ‘adam’ optimizer and ‘sparse\_categorical\_crossentropy’ loss. It had a training for 10 epochs with batch size of 25% took place employing %of the data for validation.  
  
Model Evaluation:  
We showed the training and validation loss, loaded test images and labels into a dataset deployment cookbook for inference by using Amazon SageMaker Processing jobs, evaluated the trained model performance on test set.

**Conclusion:**  
**These projects step by step offered extensive data exploration, preprocessing, model creation and evaluation in various machine learning scenarios.**





# **Project Link: https://github.com/Mosyexp/machineproject**

# References:

* Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830.
* McKinney, W. (2010). Data Structures for Statistical Computing in Python. Proceedings of the 9th Python in Science Conference, 51-56.
* Chollet, F., et al. (2015). Keras. GitHub Repository. <https://github.com/keras-team/keras>
* OpenCV Library. <https://opencv.org/>
* NumPy Documentation. <https://numpy.org/doc/>
* Pandas Documentation. <https://pandas.pydata.org/docs/>
* Matplotlib Documentation. <https://matplotlib.org/stable/contents.html>
* Seaborn Documentation. <https://seaborn.pydata.org/>
* GitHub Repository - Machine Learning Projects: https://**github.com/Mosyexp/machineproject**

**NAMES + ID:**

42010045 – Mostafa Hisham Elsayed

42010626 - Mina Timothawes

42010242 - Kerolos Nabil