**ANALYSIS AND PREDICTION OF SUPERSTORE SALES USING MACHINE LEARNING**

ECE 537 Data Mining, Winter 2024

Aswin Gunasekaran – MS Data Science (CIS)- Feature Engineering,Model Building and Evaluation,Tableau visualization

Vidarshana Govilesh – MS Data Science (CIS)- Preprocessing,Model Building and Evaluation,Tableau visualization

**INTRODUCTION**

**Topic**

This project revolves around analyzing and forecasting sales of supermarket.Utilizing historical data can go a long way in identifying important patterns and trends and help understand what works best for them.By leveraging these insights,the company can make more informed decisions and make strategies that can significantly increase the sales of their store while enhancing customer satisfaction which ultimately leads to good profit.

**Background**

Superstores confront many difficulties in the competitive retail market of today, such as controlling huge inventories, reaching a wide range of consumers with effective marketing, and adjusting to shifting consumer behavior. Because traditional approaches cannot properly comprehend or evaluate complex data trends, they frequently fail to predict sales. With the use of cutting-edge data analysis methods, this study seeks to enhance superstores' ability to estimate future sales. We intend to improve the precision of sales forecasts by utilizing machine learning and contemporary statistical models to analyze historical sales data. Superstores will be able to better meet consumer needs, manage their inventory, and develop marketing plans as a result. Superstores will be able to make better decisions that boost operational effectiveness and boost earnings with the help of improved forecasting. This project supports the current retail trend of data-driven decision-making.

**Brief Summary**

The core objective of this initiative is to engineer a predictive model capable of accurately forecasting superstore sales. We want to determine the most accurate and dependable algorithm for this prediction by conducting experiments with several models.

This project is to analyse a sales of a superstore and to find an efficient model that can analyse and predict the sales of the store.The dataset contains several attributes including product name, segment, country,region, category, and subcategory. The dataset contains around 18 columns in total and we perfomed analysis using Tableau to get insights about the data which helped us to preprocess and handle the dimensions. Several base models were employed followed by hyperparameter tuning and then employed Ensemble techniques to enhance the model performance. LightGBM emerged as the best model for analysis and predicting sales data.

**METHODS USED**

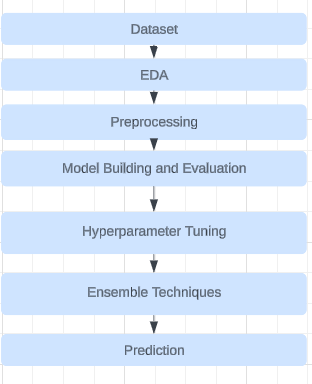
**Data Mining Technologies:**

This project leveraged a suite of key Python libraries to effectively perform various data mining tasks. For data manipulation, we used NumPy and pandas, which are essential for handling and preparing large datasets. Data visualization was accomplished using Tableau, providing dynamic insights through visual representations of the data. The modeling and evaluation processes, including hyperparameter tuning, were conducted using Scikit-learn, a robust library for machine learning techniques. Additionally, we implemented advanced ensemble methods with XGBoost and LightGBM, enhancing the predictive accuracy of our models. These tools together formed the technological backbone of our data analysis, allowing us to extract meaningful insights from complex retail sales data.

**Detailed Implementation**

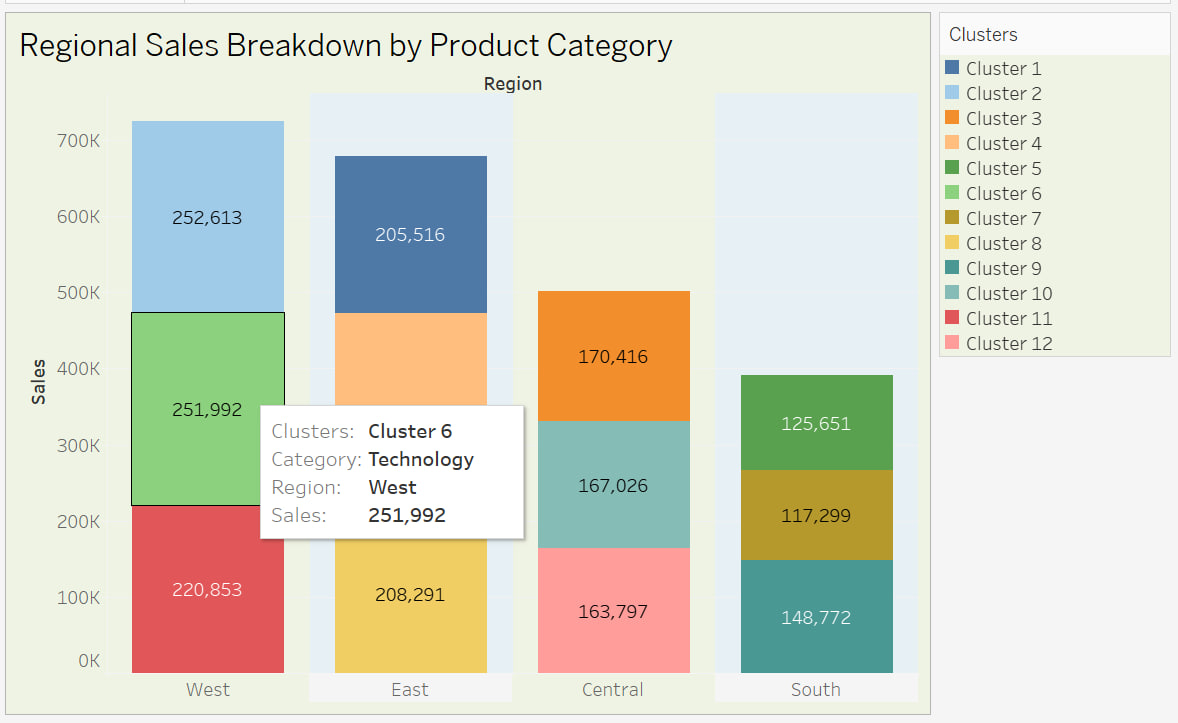
The dataset used for this project is Superstore Sales, which consists of numerical and categorical variables with target varaiable as Sales. The dataset contains 18 columns and around 9800 entries of data.The attributes include RowID, Order ID, Order Date, Ship Date, Ship Mode, Customer ID, Customer name, Segment, Country, City, State, Postal code, Region, Product ID, Category, Sub-category, Product name, Sales.

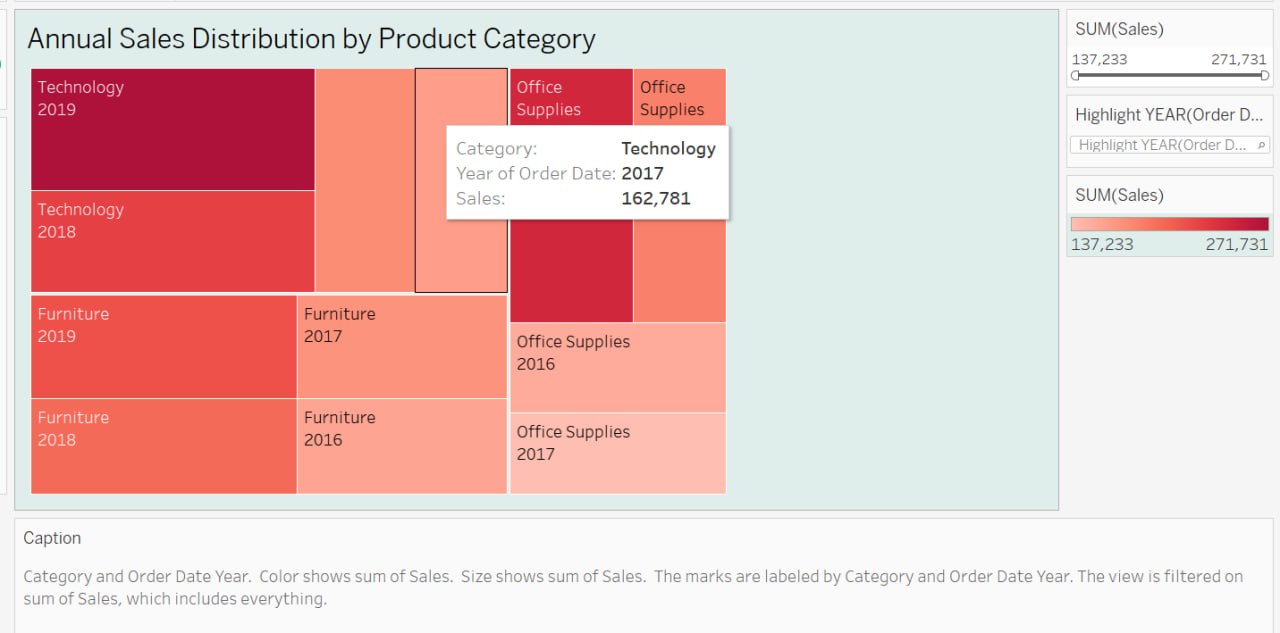
**Workflow**

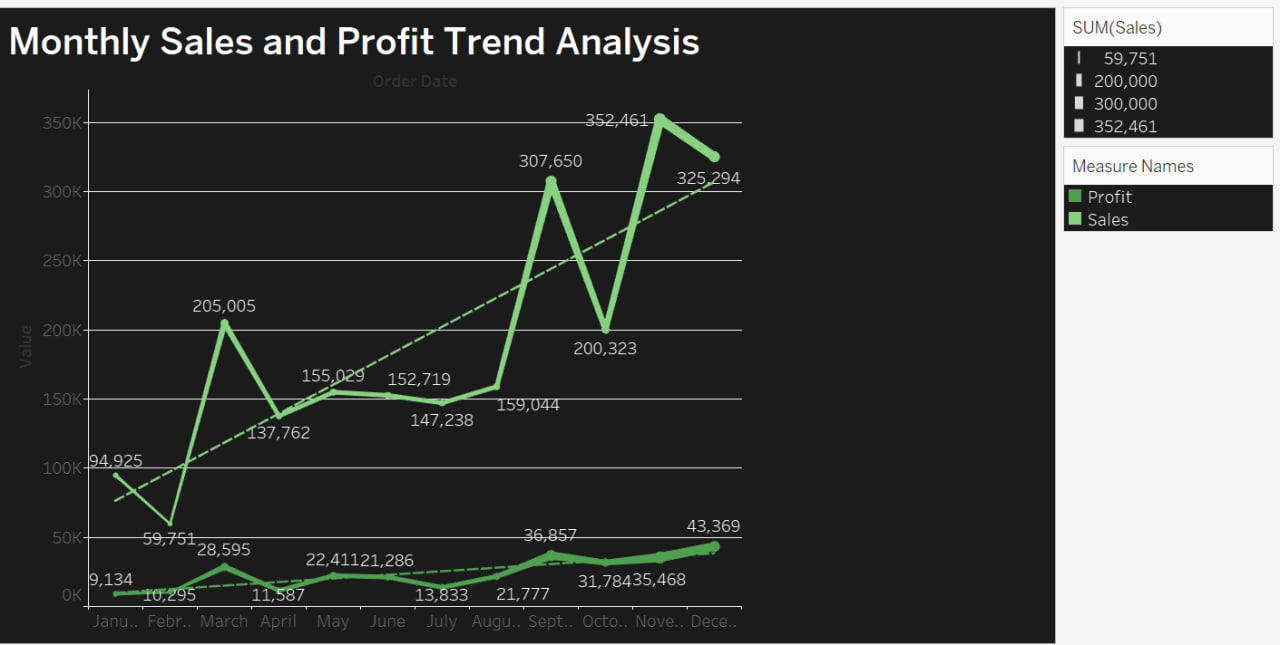
****

**Exploratory Data Visualization using Tableau**

Initial data visualization helped in understanding the distribution and impact of various features on sales.Here are some of the Visualization charts we created







**Data Preprocessing**

Firstly we used df.info()` and `df.describe()` to get an overview and statistical summary of the data.We then dropped unnecessary columns based on the domain knowledge and then handled missing values.This was then followed by converting orderdate and shipdate to datetime format and then checked for duplicate rows which were none.Then we addressed the outliers for the target variable and used log transformation for the same.Finally encoded the categorical variables.

**Feature Engineering and Splitting data**

As our dataset had a lot of attributes,we employed Principle Component Analysis technique to extract features from our data.It is known to reduce feature redudency and also have results as a normalized combination of the original data.For train test split ,we had a split ratio of 80 and 20.That is 80% data for training and 20% data for testing our models.The data was also scaled.

**Model Building and Evaluation**

For this project we used several base models like Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regressor, Random Forest Regressor, and Support Vector Regression (SVR). And the evaluation metrics used here to assess our model performance was Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R²).

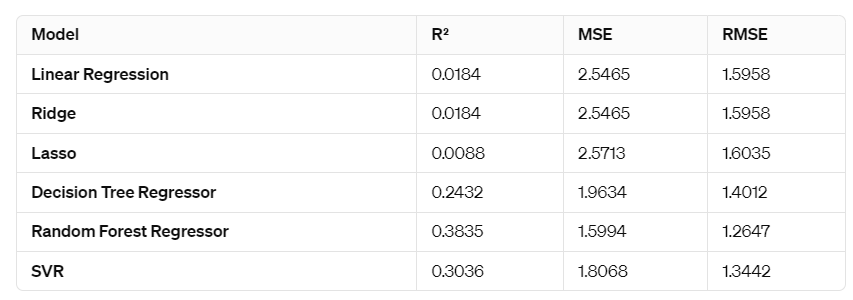
**Model Optimization and Advanced Techniques**

To Further enhance our model performance ,the best performing base model was hyperparameter tuned using the Gridsearch CV method and the best paramaters were found.Using them ,we checked the models performance to see a increase in the metrics.Later we also employed a few ensemble techniques such as LightGBM,Stacking,XGboost to try to enhance the accuracy of our prediction.

**Prediction**

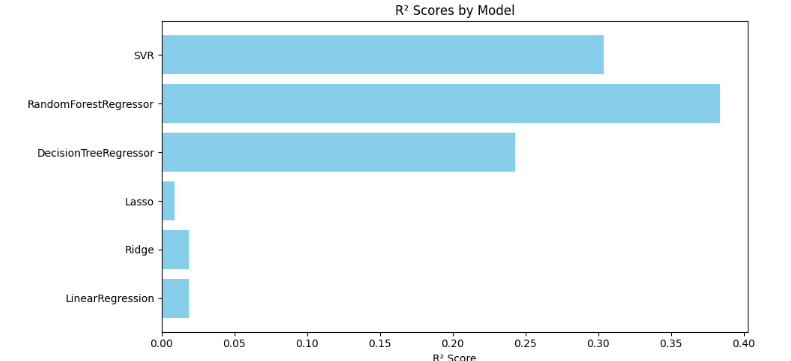
Finally as the last step of our project,we choose the best performing model overall, and we tried to predict the sales of one row.And then compared the results to actual value of sales.If the results were close to each other,then we can conclude that the model trained was performing well.

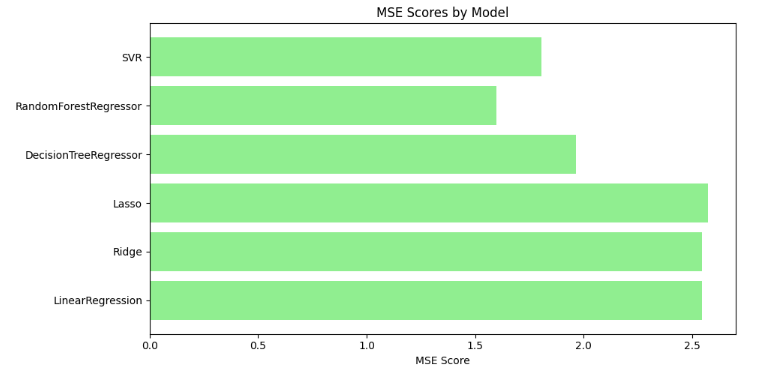
**RESULTS DISCUSSION**

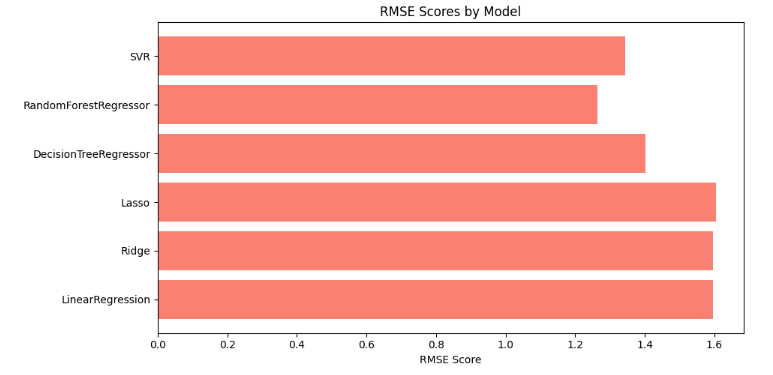
****

When discussing about these three evaluation metrics ,R-sqaured error which discusses the models variance, has to be closer to 1.The Mean Sqaured Error (MSE) and (Root Mean Squared Error)RMSE scores are the metrics what would show us how close our predicted value is to the actual value.These values have to be closer to 0.

In our project,the linear and Ridge regression models have almost the same values of Rsqaured,MSE and RMSE. The MSE and RMSE are relatively high, indicating poor prediction accuracy.While the Decision Tree Regressor shows some improvement over the linear models with an R² of 0.2432, suggesting it's able to capture more complexity in the data,Lasso perfoms slightly worse than them all.SVR with an R² of 0.3036, it shows that it can capture a moderate amount of variance in the target variable.Random Forest regressor stands out with the highest R² of 0.3835, showing a better fit than the other models. Its MSE and RMSE are the lowest among all models.

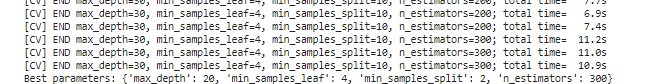




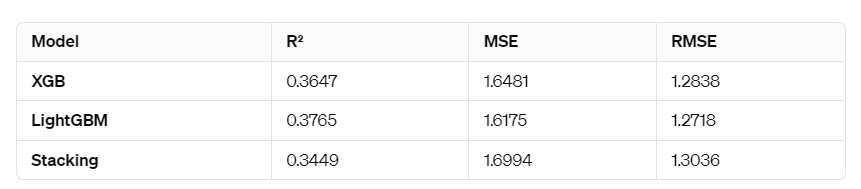


**Hyperparameter Tuning**

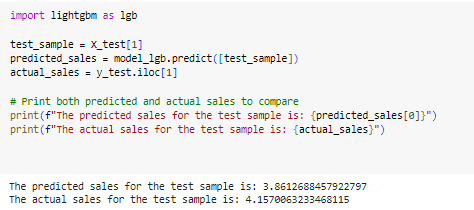
We're tuning RandomForestRegressor's hyperparameters using GridSearchCV, finding the best combination to minimize mean squared error. After fitting on training data, we print the optimal parameters and corresponding score.

****

**Ensemble Results**

Based on the results, LightGBM appears to be the best model among the three, with the highest R-squared value of 0.3765 and the lowest RMSE of 1.2718. This suggests that LightGBM not only explains a greater proportion of the variance in the target variable but also has the smallest average error in predictions. The close performance of XGB, with slightly lower metrics, indicates it's also a strong model. Stacking, while still robust, falls slightly behind in both explanatory power and prediction accuracy. 

**Prediction**



The model's performance appears decent as it is close to the actual sales figure, with a slight underestimation. This level of variance is relatively small and suggests that the model is capturing the underlying patterns in the data fairly well. However, there's always room for improvement, and further analysis could help refine the model to enhance its predictive accuracy even more.

**EXPERIMENTS**

We visualized the data generated from ImageDataGenerator and displayed it using the matplotlib library.

Several images of the inside of a person's body

Description automatically generated

**Fig1**: Visualization of train, test and validation images

**Methodology**

In this project we intended to first apply the preprocessing technique to increase the data count by utilizing the Data Augmentation Technique and generate new data using features like rotation range, rescale, horizontal flip, fill mode. After this, we used the deep learning algorithms to train and test our model. The parameters of model were tuned for each and a comparison of the different optimizers and results for each epoch was also compared. below is a summary of each model:

1. Convolutional Neural Network (CNN)

It is a deep learning algorithm that is the base for all the other pre-trained models. It is predominantly used for image classification.

1. Inception-V3

It is a deep convolutional neural

network (Deep CNN) which is usually

used for image classification whose

architecture has 48 different layers.

1. VGG16- It is also a convolution neural network model which is 16 layers deep and used for object detection.
2. Resnet50 – It is a CNN architecture that specialized in solving issues like vanishing gradient and used to solve complex problems.
3. DenseNet121- a pretrained model that has been trained on the ImageNet database, showing better feature use efficiency.

After the comparative study between the different deep learning models after hyper parameter tuning, evaluation metrics are compared between each of them and final conclusion was made that DenseNet outperformed all the other Models.

After getting the results of each model, we decided to move forward by utilizing the low performing models for an ensemble technique where the two models are combined to make a new model with better prediction.

Finally predicted the classes of image on new unseen data using the best performing model.

**Experiment**

We experimented with five different pre-trained models and ensemble model, using data split in the ratio of 8:1:1 (80% train, 10% test, 10% validation). The models were evaluation by confusion matrix and classification report with attributes like accuracy, precision, recall, F1 score and support. Out of the six different models (including ensemble model), the best-performing model turned out to be DenseNet121.

|  |  |
| --- | --- |
| Options |  |
| Optimizer | Stochastic Gradient Descent |
| Batch Size | 32 |
| Input Image Size | 224,224,3 |
| Epochs | 30 |
| No of features | 8 |

Table1: Parameters of Model Architecture

**Model Evaluation**

We have used matplotlib to visualize the difference in between two evaluation techniques namely confusion matrix and classification report.

Confusion Matrix: - It is a measurement tool used to evaluate performance in machine learning. It presents a tabular form which depicts the actual predictions vs model predictions. The parameters used in the matrix are as follows: -

Another evaluation metric used is the classification report. This metric provides a short summary of the model’s performance using a few parameters. The parameters are as follows: -  
  
Precision – It is defined as the measure of True Positives divided by Total predicted positive values.

Recall – It is defined as True Positives divided by Total actual positive values.

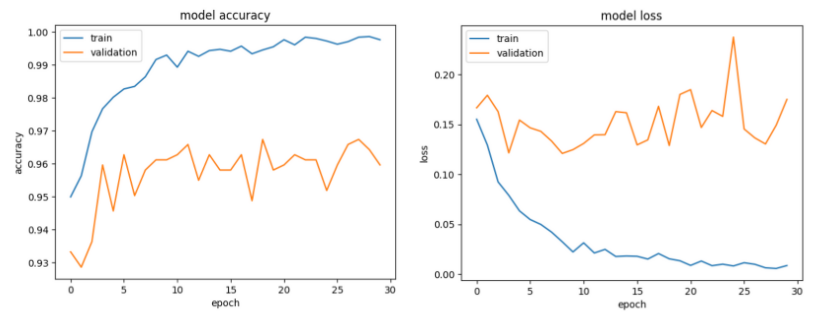
F1 Score – It is defined as the harmonic mean of the precision and recall metrics provided by the report.

Support – It is defined as the number of outcomes from each class present in the data.

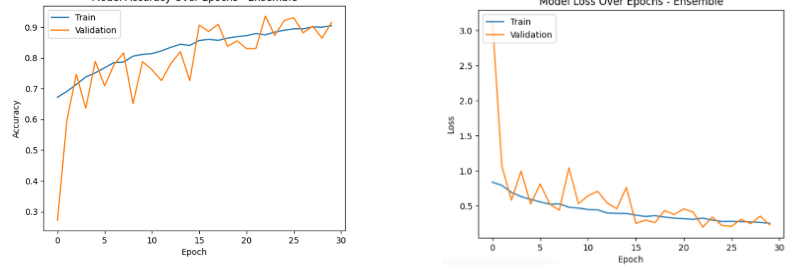
Accuracy- It is defined as the measure that describes how accurate or correct the predictions actually are.

The Training set and the validation set were monitored and visualized by plotting an ‘Epoch vs Accuracy & Epoch vs Loss’ Graph using the matplotlib library.

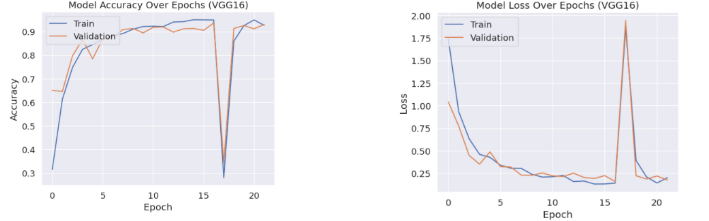
The model evaluation is as follows: -



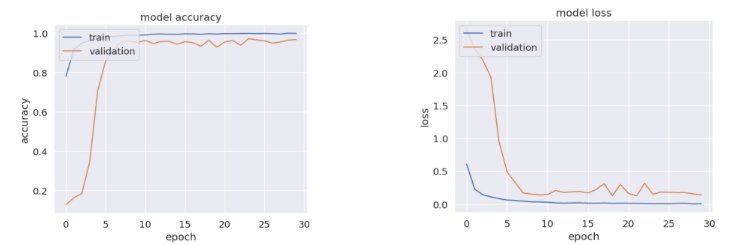
**Fig2**: Epoch vs Accuracy, Epoch vs loss graph for InceptionV3



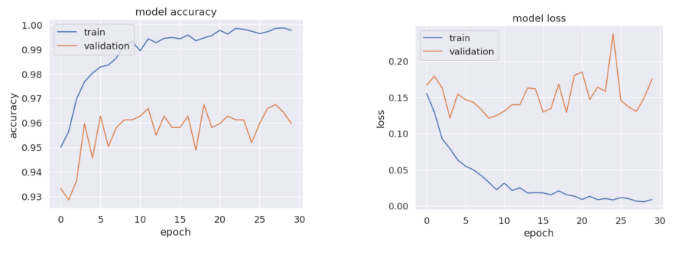
**Fig3**: Epoch vs Accuracy, Epoch vs loss graph for CNN



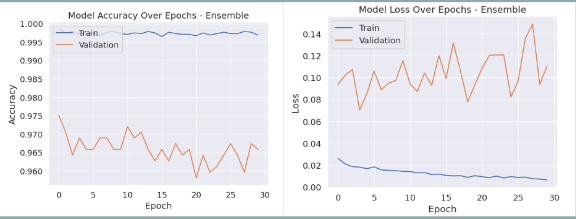
**Fig4**: Epoch vs Accuracy, Epoch vs loss graph for VGG16



**Fig5**: Epoch vs Accuracy, Epoch vs loss graph for ResNet50



**Fig6**: Epoch vs Accuracy, Epoch vs loss graph for DenseNet121



**Fig7**: Epoch vs Accuracy, Epoch vs loss graph for DenseNet121

|  |  |  |
| --- | --- | --- |
| OPTIMIZER | ACCURACY | LOSS |
| RMSPROP | 92.24 | 0.3838 |
| ADAM | 95.19 | 0.1732 |
| SGD | 97.05 | 0.1831 |

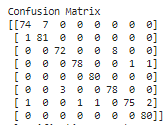
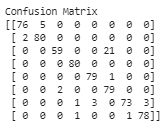
Table2: Comparison of different optimizers

**Results**

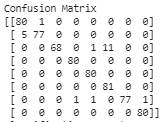
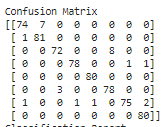
The different models along with the ensemble method were evaluated based on their accuracy and the loss. The table below gives information regarding the same.

|  |  |  |  |
| --- | --- | --- | --- |
| MODEL | ACCURACY | LOSS | TIME |
| CNN | 84.47% | 0.422 | 31mins |
| INCEPTIONv3 | 97.05% | 0.125 | 28mins |
| VGG16 | 93.17% | 0.1498 | 27mins |
| RESNET50 | 96.74% | 0.1831 | 27 mins |
| DENSENET121 | 97.83% | 0.0729 | 25mins |
| ENSEMBLE | 97.52% | 0.0741 | 30mins |

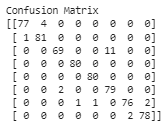
The confusion matrix is obtained as follows:

 ****

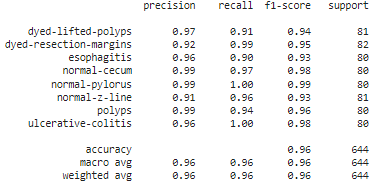
**Fig8**: Confusion Matrix (a) InceptionV3, (b) VGG16

** **

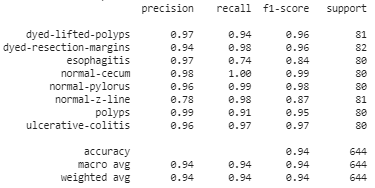
**Fig9**: Confusion Matrix (a) Resnet50, (b) Densenet121



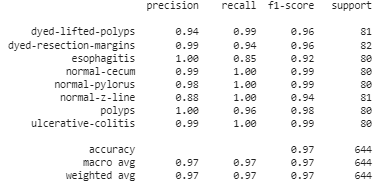
**Fig10**: Confusion Matrix (a) Ensemble Method (VGG16 and Resnet50)



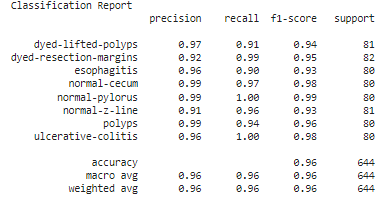
**Fig11:** Classification Report for InceptionV3



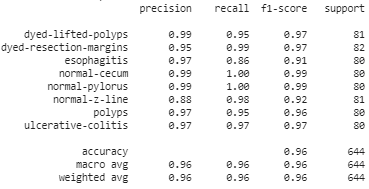
**Fig12:** Classification Report for VGG-16: -



**Fig13:** Classification Report for ResNet50:-



**Fig14:** Classification Report for Dense Net



**Fig15:** Classification Report for Ensemble-

Conclusion

This study clearly shows that post augmentation and preprocessing of the data, out of all the models, DenseNet121 proved to be the superior model post hyper parameter tuning. For minimal epochs and freezing a few layers, the obtained accuracy is 97.8%. Hence we used this model to make predictions on unseen data and the two images used to test were correctly classified into their class by the DenseNet model. Future work would focus on using infused architectures to enhance precision and prediction for all the classes present in the dataset.

**References**

[1] Patel, Zeel & Jaiswal, Rupesh & Potdar, Girish & khodaskar, M.R.. (2022). A SURVEY PAPER ON BIG DATA ANALYTICS IN SALES AND MARKETING. 10. c420-c428.

[2] J. S. Kiran, P. S. V. S. Rao, P. V. R. D. P. Rao, B. S. Babu and N. Divya, "Analysis on the Prediction of Sales using Various Machine Learning Testing Algorithms," 2022 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2022, pp. 1-6, doi: 10.1109/ICCCI54379.2022.9740949. keywords: {Training;Analytical models;Machine learning algorithms;Linear regression;Organizations;Predictive models;Prediction algorithms;Sales;linear regression;Random forest;XG Booster;Bayesian Regression}

[3] Pavlyshenko BM. Machine-Learning Models for Sales Time Series Forecasting. Data. 2019; 4(1):15. <https://doi.org/10.3390/data4010015>

[4] Bajaj, P., Ray, R., Shedge, S., Vidhate, S., & Shardoor, N. (2020). Sales prediction using machine learning algorithms. *International Research Journal of Engineering and Technology (IRJET)*, *7*(6), 3619-3625.

[5] Cheriyan, Sunitha, Shaniba Ibrahim, Saju Mohanan, and Susan Treesa. "Intelligent Sales Prediction Using Machine Learning Techniques." In 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE), pp. 53-58. IEEE, 2018.

**Responsibilities**

Lohit Arun Saravanan: Data preprocessing, Augmentation and Visualization, InceptionV3

Aswin Gunasekaran: CNN, DenseNet121, Model Evaluation,Ensemble Technique

Vidarshana Govilesh: VGG16, Resnet50, Hyperparameter Comparison, Prediction