Case Study Summary Session 2

MIT-ADSP

Topics

1. Practical Data Science
    1.1. Decision Tree and Random Forest
    1.2. Time Series
2. Deep Learning
    2.1. Artificial Neural Networks
    2.2. Convolutional Neural Networks
3. Recommendation Systems
    3.1. Part 1 - Rank-Based, Similarity-Based & Matrix Factorization
    3.2. Part 2 - Clustering-Based & Content-Based
4. Appendix

Practical Data Science

Decision Trees and Random Forest

Topics

Employee Attrition Case Study

Objective and dataset	• McCurr Healthcare Consultancy is an MNC that has thousands of employees spread out across the globe • The Head of People Operations wants to bring down the cost of retaining employees • Identify the factors that drive attrition and build a model to predict if an employee will attrite or not • The data contains employee information like demographic details, work-related metrics and attrition flag
Approach	• Load the dataset and drop unnecessary columns • Perform univariate and bivariate analysis. Check correlation among numerical variables • Encode the categorical variables and split the data into training and testing data in 70:30 ratio • Build the model decision tree and random forest models and analyze the performance on the training and testing data • Visualize the decision tree and check the feature importance of both the models • Try to improve the model performance using hyperparameter tuning with GridSearchCV
Key Findings	• The Decision Tree model with default parameters is overfitting the training data • Tuning the decision tree model reduced overfitting but recall on the test data decreased significantly • The Random Forest classifier is overfitting the data as well. Recall on the test data is about 79% • The tuned random forest model is also comparatively overfitting the training dataset, but it shows a very good performance on the test dataset. The recall for the tuned model has improved from 79% to 83% with a small decrease in precision • The feature importance plot for tuned random forest model suggests that OverTime, MonthlyIncome, Age, TotalWorkingYears, and DailyRate are the most important features
Conclusions and Recommendations	• The tuned random forest model is the best model with nearly ~83% recall on the test data. The company can use this model to know beforehand which employee is going to attrite and act accordingly • The organization should manage their work more efficiently so that employees don't have to work overtime and can manage to have a work-life balance, or failing this, the company could provide some additional incentives to employees who are working overtime • The company should make sure that all its employees are compensated at least based on industry standards • The company should also keep track of the hourly rate or the daily rate, so that when the employees need to stay overtime for extra work they are well compensated for the same

Practical Data Science

Time Series

Topics

Crude Oil Production Forecasting Case Study

Great Learning
POWER AHEAD

Objective and dataset

- The world economy relies heavily on hydrocarbons, particularly oil, for the provision of energy required in transportation and industry
- We need to build a time series forecasting model using the AR, MA, ARMA & ARIMA models in order to forecast oil production
- The `Crude Oil Production by Country` dataset contains the yearly oil production of 222 countries from 1992 till 2018, but for simplicity, we will use only one country i.e. Russia to forecast its future oil production

Approach

- Load the dataset, extracting the oil production for Russia and visualize the time series
- Split the dataset in train and test data. Keep the last 7 years as the test data and test the stationarity of the training data
- Perform second order differencing to make the data stationary and visualize the demposed data
- Visualize acf and pacf plots
- Build the model for different values of p and q using different algorithms - AR, MA, ARMA, and ARIMA and visualize the output and calculate the AIC and RMSE scores
- Fit the best model on the complete data and forecast the series for the next 7 years
- Forecast the series for last 12 months (from April 2020 to March 2021) and visualize the same
- Additional Model - Auto ARIMA and visualize the plot diagnostics

Key Findings

- The oil production of Russia was declining from early 1990s to late 1990s but has been increasing almost constantly since then
- The seasonal and residual components are zero, as the time series has yearly frequency
- As none of the acf or pacf plots tails off or cuts of after any lag. It implies that the series follows an ARMA or ARIMA model
- The ARIMA(0, 2, 0) is the best model as compared to others, as it has comparable AIC as compared to other models, but it has the least RMSE. This also makes sense, from the ACF and PACF plots that we observed
- The forecast for the next 7 years indicates that, according to the historic data, the oil production is going to constantly increase for Russia
- For the current data, we got the same parameters by auto_arima model as well

Conclusions and Recommendations

- We have built different types of models using search for the optimal parameters for each. We have compared all the models based on the evaluation metrics AIC and RMSE.
- The AIC for all the models are approximately the same i.e. there is no significant difference in the AIC values for all the models. But we can see significant difference in some of the models in terms of RMSE. So the choice of model is more dependent on RMSE for the current data.
- Overall, the model ARIMA(0, 2, 0) has given the best results and we have used the same to forecast the oil production for Russia.

Deep Learning

Artificial Neural Networks

Topics

Predicting Chances of Admission

Objective and dataset

- A classification model can be used to help students in shortlisting universities based on their profiles.
- Build a classification model using neural networks to predict a student's chances of getting admission into UCLA.
- The dataset contains several parameters which are considered important during the application for Master's Programs. For example, GRE and TOEFL scores, SOP and LOR strength, GPA, etc.

Approach

- Load the Libraries and the dataset.
- Convert target variable to categorical variable, drop unnecessary columns and perform some visualizations to extract insights.
- Perform one-hot encoding on categorical variables and scaling on numerical variables as a preprocessing step for the neural network.
- Build three different models and compare the results among them.
- The first model consists of 2 hidden layers and the output layer. The second model consists of 3 hidden layers and an output layer. And the third model consists of 4 hidden layers and the output layer.
- The models are also different from each other in terms of other hyperparameters, for example, number of neurons in each hidden layer, activation function, optimizers, etc.

Key Findings

- We observed that the majority of students who were admitted have GRE score greater than 320, TOEFL score greater than 105, and SOP of 4 or higher. Also, the CGPA of students getting admission to UCLA is higher.
- Increasing the number of hidden layers and changing the activation function for hidden layers from relu to tanh has improved the model performance.
- Among all three models, model 3 gives the best performance where the test accuracy is coming out to be 95%.

Deep Learning

Convolutional Neural Networks

Topics

Image Classification on CIFAR 10 Dataset

Great Learning
POWER AHEAD

Objective and dataset

- Build a multi-class classification algorithm to predict 10 different classes of the CIFAR-10 dataset using Convolutional Neural Networks and Transfer Learning.
- The CIFAR-10 dataset consists of 60,000 low-resolution (32x32x3) color images in 10 classes (airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks) with 6,000 images per class.
- There are 50,000 training images and 10,000 test images.

Approach

- Load necessary libraries and the CIFAR10 dataset from Keras.
- Convert NumPy arrays to images and visualize some random images.
- Normalize the feature inputs.
- Build 4 different models and compare the results among them.
- The first 3 models will differ from each other in terms of convolutional layers, dropout values, and other hyperparameters.
- Apply Transfer Learning for the 4th model and use VGG16 as the pre-trained model.

Key Findings

- Dropout layers seem to reduce the overfitting. Model 3 gives the best performance among the first 3 models where the number of convolutional layers is more but the total number of trainable parameters is comparatively less due to the addition of max-pooling layers.
- We get the best validation accuracy from model 4 i.e from the transfer learning model without actually training any of the convolutional layers.
- The model gives about 78% accuracy on the test data which is comparable to the accuracy of the validation data.
- The recall has a high range which implies that the model is good at identifying some objects while poor at identifying some other objects. For example, the model can identify more than 90% of ships but can identify only ~63% of dogs.
- The model is majorly confused between cats and dogs. This implies that the model might be focused on features related to shapes and sizes but not deep features of objects that can help to distinguish between objects like cats and dogs.
- Consequently, precision also has a high range with the 'cat' class having the least precision.
- The highest precision is for 'horse' which implies that the model can distinguish horses from other objects.

Recommendation Systems

Rank-Based, Similarity-Based & Matrix Factorization

Topics

Movie Recommendation Systems Part 1

Great Learning
POWER AHEAD

Objective and dataset

- Build a recommendation system to recommend relevant movies to users based on their historical interactions.
- Building three types of recommendation systems: Knowledge/Rank-based recommendation system, Similarity-Based Collaborative Filtering, Matrix Factorization Based Collaborative Filtering.
- We have two datasets. The 'rating' dataset contains attributes like userId, movieId, rating, timestamp. The 'movies' dataset contains attributes like movieId, titles, genres.

Approach

- Load necessary libraries and import the data. Drop unnecessary columns from the dataset.
- Explore the dataset using some plots and basic data analysis techniques.
- Build a Rank based recommendation system to find the top 'n' movies for a recommendation based on the average ratings of movies.
- Introduce the Surprise library in python to build Similarity based(Both user-user and item-item similarity) collaborative filtering. Used the same to predict the rating by an user for both a seen and an unseen movie.
- Implement the matrix factorization approach with the SVD (SIngular Value Decomposition) algorithm to predict the rating given by users. Create a function to recommend movies to the users based on predicted ratings for each movie.
- Tuned all the models using GridSearchCV
- Used precision@k, recall@k, F1-Score@k and RMSE as metrics for model performance evaluation

Key Findings

- Provided movie recommendations to users using all three algorithms.
- For the user-user similarity based collaborative filtering model the F_1 score (Calculated using precision@k and recall @k) is found to be maximum.
- We can further try to improve the model performance using hyperparameter tuning

Recommendation Systems

Clustering-Based & Content-Based

Topics

Movie Recommendation Systems Part 2

Great Learning
POWER AHEAD

Objective and dataset

- Online streaming platforms like Netflix have plenty of movies in their repositories and we can build a recommendation system to recommend relevant movies to users based on their historical interactions, this would improve customer satisfaction and hence improve revenue.
- Building various recommendation systems: Clustering-based recommendation system, Content-based collaborative filtering.
- We have 3 datasets. The 'rating' dataset contains attributes like userId, movieId, rating, timestamp. The 'movies' dataset to get the title of the movies, which contains attributes like movieId, titles, genres, and the 'tags' dataset contains attributes like userId, movieId, timestamp, tag: Brief comments about the movie.

Approach

- Load necessary libraries and import the data. Merge both the rating and movies dataset to get the title and rating of each movie in a single dataframe.
- Perform some basic data analysis techniques to explore the dataset.
- Used the Surprise package in python to build the clustering based recommendation system using the CoClustering algorithm with the detailed explanation of the algorithm too.
- For model evaluation precision@k and recall@k are used (along with RMSE) with their detailed explanation. Using them, F_score is computed as the final performance metrics for the model.
- To build the content-based recommendation system use the tags dataset too, as we would be using the text features i.e. reviews to find out similar movies. Merge all three datasets on movieId. Keep only four columns - userId, movieId, rating, and text (title, genres, and tag). Use NLTK and scikit-learn library to extract features using the TF-IDF technique. Create the function to find the most similar movies to recommend for a given movie.

Key Findings

- How two models are different from each other and what kind of data is needed to build each of these two recommendation systems.
- Provided movie recommendations to users using all clustering-based and content-based algorithms.

Appendix

Topics

UBER Case Study

Objective and dataset	• Find out the different factors that influence pickups and the reason for such influence • To identify some key insights that Uber management can take reference from to capitalize on fluctuating demand. • The Uber dataset contains various information about weather, location, and no. of pickups.
Approach	• Load the dataset, fill in the missing values and drop unnecessary columns. • Perform univariate and multivariate analysis. Explore categorical variables and check correlation among numerical variables. • Draw meaningful conclusions from the different plots.
Key Findings	• Uber cabs are most popular in the Manhattan area of New York. • Contrary to intuition, weather conditions do not have much impact on the number of Uber pickups. • The demand for Uber increased steadily over the months (Jan to June). • The rate of pickups is higher on the weekends as compared to weekdays. • People use Uber for regular office commutes. The demand steadily increases from 6 AM to 10 AM, then declines a little and starts picking up till midnight. The demand peaks at 7-8 PM. • New Yorkers trust Uber taxi services when they step out to enjoy their evenings.
Conclusions and Recommendations	• Manhattan is the most mature market for Uber. Brooklyn, Queens, and Bronx show potential. • There has been a gradual increase in Uber rides over the last few months and we need to keep up the momentum. • Ridership is high at peak office commute hours on weekdays and during late evenings on Saturdays. Cab availability must be ensured during these times. • The demand for cabs is highest on Saturday nights. Cab availability must be ensured during this time of the week. • Procure data for fleet size availability to get a better understanding of the demand-supply status and build a machine learning model to accurately predict pickups per hour, to optimize the cab fleet in respective areas. • Procure more data on price and build a model that can predict optimal pricing.

Inferential Statistics - Medicone Dose Testing Case Study

Great Learning
POWER AHEAD

Objective and dataset

- Medicone, a pharmaceutical company, has manufactured the sixth batch of COVID-19 vaccine doses.
- They want to understand the quality of the vaccine with the help of some voluntary trials.
- Find the quality and effectiveness of the vaccine so that the company can plan for the next batch of doses.

Approach

- Read the description of the problem and find some important properties from the results of previous studies.
- Plot the probability distribution for doses and find the results from PMF and CDF.
- Load the dataset and use some visualization to find the data distribution type.

Key Findings

- The probability that the vaccines will not do a satisfactory job is very low.
- The probability that exactly 3 doses will not do a satisfactory job is 1.25%, and the probability that at most 3 doses will not do a satisfactory job is 1.73%.
- The probability that at least 30 doses will not do a satisfactory job out of 200 doses is ~0.3%.
- 95% of the time, the mean effective time of doses will be between 12.09 hours and 14.79 hours.

Hypothesis Testing - Mobile Internet Case Study

Great Learning
POWER AHEAD

Objective and dataset	• ExperienceMyServices reported that a typical American spends an average of 144 minutes (2.4 hours) per day accessing the Internet via a mobile device with a standard deviation of 110 minutes. Test the validity of this statement. • Dataset consists of 30 samples and only the value of time spent per day accessing the Internet . • Find out if there is enough statistical evidence to conclude that the population average time spent per day accessing the Internet via mobile device is different from 144 minutes?
Approach	• Define null and alternate hypotheses and decide the significance level. • Identify the test statistic - z-statistic or t-statistic. • Calculate the p-value using the test statistic. • Decide whether to reject the null hypothesis or not based on the test statistic.
Key Findings	• At 5% significance level, we do not have enough statistical evidence to prove that the average time spent on the Internet is not equal to 144 minutes.

Education and Air Pollution Case Study

Great Learning
POWER AHEAD

Objective and dataset

- Reduce the number of features using dimensionality reduction techniques like PCA and t-SNE
- Apply techniques on two datasets and interpret/visualize the results
- Education dataset containing information on educational institutes in USA
- Air pollution dataset containing information on major pollutants and meteorological levels of a city

Approach

- Load the education dataset and perform basic univariate analysis
- Fixed percentage values greater than 100 for two columns
- Check correlation among numerical variables and scale the data
- Reduce dimensions using PCA and t-SNE and visualize the data in 2 dimensions
- Load the air pollution dataset and drop serial number and date columns
- Check and impute missing values
- Reduce dimensions using PCA and t-SNE and visualize the data in 2 dimensions

Key Findings

- In education dataset, reduced the number of features by ~76% (from 17 to 4) using PCA with 30% loss in variance
- Represented each principal component as a linear combination of original features
- PC1 captures attributes that define premier colleges with high quality of students entering them and higher accomplishing faculty that is teaching there. They also seems to take rich students from all over the country.
- PC2 captures attributes that generally define non-premier colleges that are comparatively easier to get admissions into
- PC3 is related to financial aspects and low values of student faculty ratios
- PC4 captures attributes that define colleges with lack the highly educated faculty but it is comparatively easier to graduate from there
- No meaningful pattern observed using t-SNE. The data is clustered together with some outliers
- In air pollution dataset, reduced the number of features by ~80% (from 25 to 5) using PCA with 30% loss in variance
- Visualized the data in dimensions using t-SNE
- The data forms 4 groups where Group 1 represents hot and humid areas, Group 2 represents developing urban areas, Group 3 represents the developed urban areas, and Group 4 represents the industrial areas

Caviar Investigation Phases Case Study

Great Learning
POWER AHEAD

Objective and dataset

- A time-varying criminal network that is repeatedly disturbed by police forces from 1994 to 1996 in eleven phases
- The network consists of 110 (numbered) players. Players 1-82 are the traffickers. Players 83-110 are the non-traffickers (financial investors; accountants; owners of various importation businesses, etc.)
- Understand, create and visualise the data in phases and figure out important nodes across phases

Approach

- Read the data and understand the structure of data
- Put the data into a graph and visualize the graph
- Identify the important nodes from the visualization
- Calculate the centrality measures (Degree, Eigen, Betweenness, Closeness) and quantify the importance
- Understand the variation of node importance across phases

Key Findings

- We carried out the analysis on the network and figured out techniques to read adjacency matrices into graphs
- We later visualised the graphs, created centrality measures and identified important nodes - N1, N3, N12
- We studied and plotted the variation in the centrality of the important nodes across phases in a bid to understand the effect of disruption of the network

Country Clustering Case Study

Great Learning
POWER AHEAD

Objective and dataset

- Identify cluster of countries that are more similar to each other in terms of certain socio-economic factors
- Country dataset contains various socio-economic attributes for countries around the world
- We will not do clustering on the gdp and would rather try to understand the variation of other factors with GDP across the groups that we formed

Approach

- Load the country dataset and perform basic univariate analysis
- Check correlation among numerical variables and scale the data
- Choose best K using elbow method and Silhouette score and create cluster profiles using the K-Means clustering
- Apply K-Medoids clustering and compare the cluster profiles with K-Means clustering profiles
- Apply Gaussian Mixture clustering and compare the cluster profiles with K-Medoids clustering profiles
- Choose the number of clusters using the Dendrogram and apply hierarchical clustering
- Apply DBSCAN clustering

Key Findings

- No clear 'elbow' in the elbow plot but the Silhouette score is highest for K=3. Chosen K=3 for K-Means
- Cluster 2 has only 3 observations. It consists of outlier countries with highest imports and exports as percentage of GDP
- Cluster 1 shows traits of underdeveloped and developing countries and Cluster 3 shows traits of developing and developed countries
- Using K-Medoids, cluster 2 represents underdeveloped to developing countries, cluster 1 represents developing countries and cluster 3 represents developed countries
- The count of observations in each clusters from K-Medoids is more evenly distributed as compared to clusters K-Means
- Unlike K-Means, the clusters from K-Medoids for developed countries is much bigger but still retains the overall characteristics of developed countries
- In GMM, clusters looks very similar to the clusters from K-Medoids with one cluster of 'rich' countries, one of 'poor' and one of 'all others'. 0, 1, and 2 represents underdeveloped, developed, and underdeveloped and developed countries, resp.
- It is hard to distinguish clusters using hierarchical clustering. Therefore, we will not deep dive into the cluster profiles.
- In DBSCAN, we got 5 clusters using epsilon equal to 1. Three out of 5 clusters (0,1,& 2) seems to be way more compact across all attributes. We can explore it more to understand which type of countries it consists.
- Choice of algorithm here will depend on the context and use case. But purely based on foundations of 'what good clustering looks like', one can propose K-Medoids as it has more distinct extreme clusters of developing and underdeveloped countries.

BigMart Sales Prediction Case Study

Great Learning
POWER AHEAD

Objective and dataset

- Data scientists at BigMart have collected 2013 sales data for 1559 products across 10 stores in different cities
- Build a predictive model and find out the sales of each product at a particular store
- Provide recommendations to the BigMart sales team to understand the properties of products and stores which play a key role in increasing sales

Approach

- Load the dataset and drop ID variables - Item_Identifier, Outlet_Identifier
- Perform univariate and bivariate analysis. Check correlation among numerical variables
- Fix the data issues in the column Item_Fat_Content, impute missing values and create new feature Outlet_Age
- Prepare data for modeling and scale the training and testing data
- Build the model and select only the relevant features based on p-value (p-value<0.05)
- Check for 5 assumptions of the linear regression model
- Conclusion and Recommendations

Key Findings

- The majority of Outlet_Size is Medium, majority of Outlet_Location_Type is Tier 3, and majority of Outlet_Type is Supermarket Type 1
- The average sales are almost constant every year except 1998 where the average sales plummeted
- Age of stores does not impact the sales as different age of stores have similar distribution approximately
- After removing multicollinearity, applying log transformation on the target variable, and checking all the assumptions, the final model is giving the R-Square of 0.675
- The R-Squared and MSE on the cross validation is almost similar to the R-Squared on the training dataset

Conclusions and Recommendations

- Equation of the model implies one unit change in the variable Item_MRP, the outcome variable increases by 1.9623 units.
- On average, the log sales of stores with outlet size small is 0.5812 less than the log sales of outlet size high
- On average, the log sales of store type Supermarket 3 is more than the log sales of other types of stores.
- The management can focus on maintaining or improving the sales in large stores of supermarket type 3. And for the remaining ones we may want to make strategies to improve the sales e.g. better training for store staffs, providing more visibility of high MRP item, etc.

Employee Attrition Case Study

Great Learning
POWER AHEAD

Objective and dataset

- McCurr Healthcare Consultancy is an MNC that has thousands of employees spread out across the globe
- The Head of People Operations wants to bring down the cost of retaining employees
- Identify the factors that drive attrition and build a model to predict if an employee will attrite or not
- The data contains employee information like demographic details, work-related metrics and attrition flag

Approach

- Load the dataset and drop unnecessary columns
- Perform univariate and bivariate analysis. Check correlation among numerical variables
- Prepare data for modeling and scale the training and testing data
- Build the model using different algorithms - Logistic Regression, and KNN
- Interpret the results and print the classification metrics for the training and testing data

Key Findings

- Working overtime is the most important driver of attrition
- Attrition rate is high for sales and marketing departments
- The organization has a lower percentage salary hike and promotions are given less frequently
- Approximately 40% employees have given a poor rating on job satisfaction and environment satisfaction
- Lower job involvement leads to a higher likelihood of attrition
- Young and relatively new/inexperienced employees tend to show a higher attrition rate

Conclusions and Recommendations

- The hyperparameter tuned KNN classifier is overfitting but gives the highest recall on the training and the testing data
- The organization should manage their work more efficiently so that employees don't have to work overtime and can manage to have a work-life balance, or failing this, the company could provide some additional incentives to employees who are working overtime
- The organization could look into their incentive schemes and try to come up with better ideas to retain employees from sales and marketing departments
- The company might be able to focus on giving promotions more frequently or they could increase the annual appraisal hike. Also, a more proactive, hands-on approach may be required from the managers in the organization to avoid low job involvement