Data Analytics

AD23411

DA → Data Analytic
DAs → Data Analysis

Unit-I (notes)

*| It's all about how to turn the raw numbers to understandable Informations.

Topic to cover

*| Overview of DA (Data Analytics) ✓
*| Type of DAs (Data Analysis)
*| Steps in DAs Process
*| Data Prerequisites
*| ETL (Extract Transform Load) ✓
*| Roles Responsibilities, Skillsets of DA ✓

Data Analytics
Data Analysis

Overview of DA

*| It is the process of collecting, organizing and studying data to find or create useful Informations

*| **Importance :**

*| Helps in decision making
*| Helps in problem solving
*| Helps identify opportunities
*| Improved efficiency

*| **DA Process**

*| **Data collection** (Collecting raw data)
*| **Data cleaning** (Preparing the data)
*| **Data Analysis** : Using programming Language to Perform tasks
*| **Data visualization** (Results are visualized and analyzed)

*| Types of DA :

1) **Descriptive DA**: Analytics of past data to understand them
2) **Diagnostic DA** : It finds the reason for the past data, uses Correlation and plots graph (focuses on why?)
3) **Predictive DA**: Predicting future based on past

4| Prescriptive DA: not only predicting but finding the best solution to solve the problem.

Methods of DA:

*| Qualitative Analytics: Source: words, Images, Symbols, interviews
Methods: Content Analytics Analysis, Grounded Theory, Narrative Analytics
Goal: deriving Meaning and Context from Non-statistical Inputs

*| Quantitative Analytics: Used Numerical Data
Methods: Hypothesis Testing, Regression, Mean/Average calculations
Goal: Measuring Variables and testing theories statistically

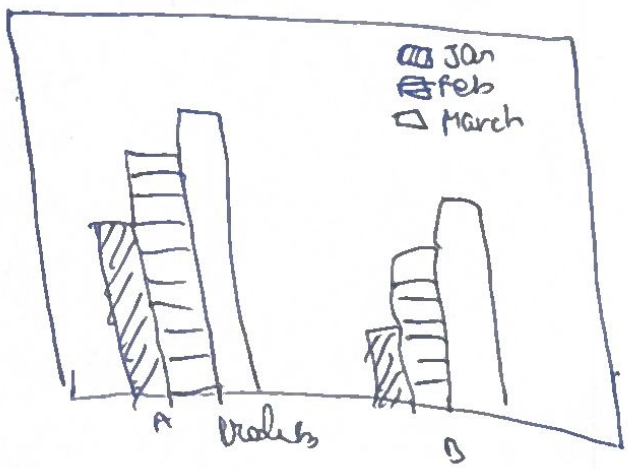
Example of Types of Data Analytics

1) Descriptive Data Analytics

code: (Pandas (pd), Matplotlib (plt))

```
data = {'Product': ['A', 'B'],
        'January': [200, 180],
        'February': [150, 210],
        'March': [300, 240]}
df = pd.DataFrame(data)
df.set_index('Products').plot(kind='bar', colormap='Viridis')
plt.title("Monthly Sales Data")
plt.xlabel("products")
plt.ylabel("Sales")
plt.show()
```

output:

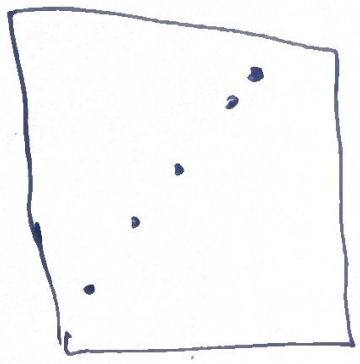


2) Diagonostic Analytics:

output:

Code: pandas as pd , seaborn as sns , matplotlib as plt

```
data = {'Ad spend' : [1000, 1500, 2000, 2500, 3000],
        'Sales' : [12000, 13000, 18000, 21000, 25000]}
df= pd. DataFrame (data)
Cor = df. Corr()
Print (Cor)
sns. scatterplot (X='Adspend', Y='Sales', data =df, color='green')
plt. title
```



Shows +ve correlation

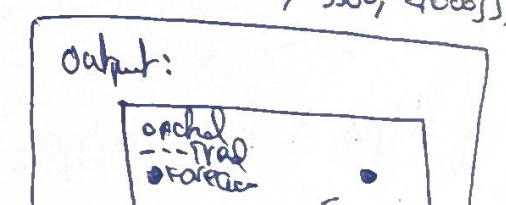
3) Predictive Analytics

pandas , matplotlib, seaborn , From ↓ sklearn. linear_model import

↓ Linear Regression

```
df = pd. DataFrame ({'Month' : [1,2,3,4,5], 'Sales': [2000, 2500, 3000, 3500, 4000]})
X, Y = df [['Month']] , df ['Sales']
Model = Linear Regression (). fit (X, y)
next month = [[6]]
Prediction = Model. prediction [next_month)[0]
Sns. set_theme (Style ="darkgrid")
plt. Scatter (X, Y, colour ='blue', label ='Actual')
plt. plot (range (1,7), Model. predict ([[i] for i in range (1,7)]), 'g--',
                                                    label ='Trend')
Plt. Scatter (next_month, Prediction, Color ='red', s=100, label = 'Forecast')
plt. title (f"Month 6 Forecast : ${Prediction :,.2f}")
Plt. legend()
plt. show()
print (f"{Prediction :,.2f}")
```

output:



4) Prescriptive DA

Code: pandas, matplotlib.pyplot as plt, seaborn as, sklearn.linear_model

```
df = pd.DataFrame({'SQFT' :[1000, 1500, 2000, 2500, 3000], 'Price' :[200k, 250k, x, y
     , :df ['SQFT'], [200000, 250000, 300000, 350000, 400000]
model = LinearRegression().fit(X, Y)
target_sqft = [[1800]]
prediction = model.predict(target_sqft)[0]
sns.set_theme(style = "whitegrid")
plt.scatter(X, Y, ...)
plt.plot(X, model.predict(X), color = 'green', label = 'Trend')
plt.scatter(...)
:
```

output:

Same as predictive code, we would just draw some conclusion, that's the difference.

Steps in Data Analysis process:

Steps:

1) Define the Problem (understand before solving)
2) Data collection (Extracting/collecting the necessary data for the process)

Code:

```
titanic = sns.load_dataset('titanic')
titanic.head()
```

output: Dataset is displayed

Shortcuts

Sns -> Seaborn
Pd -> Pandas
Plt -> Matplotlib

3) **Data cleaning**: It is the longest process, here we prepare the data for the further process i.e: Removing null, duplicates, fixing inaccuracy

code: for fillna

*) titanic['embarked'].fillna(titanic['Embarked'].mode()[0], inplace
*) df_filled['alone']= df_filled['alone'].fillna(df['alone'].mean())

4) **Analyzing Data**: Here you will visualize and find the patterns, trends behind the given data, also if there is any inaccurate value it is analyzed in this step

5) **Visualizing** Data:

*) converting to graph or any chart to easily understand the data

Similar

6) **Interpret and Make decisions**: *) This is the final step
*) Explain the result and more
*) This is 100% dependant on the user

Ex code: for accuracy

```
accuracy = accuracy_score(y_val, y_pred)
```

Data Repositories in DA

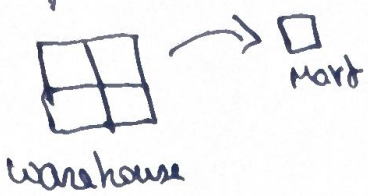
1) **Data warehouse**: It is a centralized data repository, just like real life warehouse, they are structured historical data. schema-on-write

key features:

a) Subject-oriented: organized around major bussiness subjects, rather than applications or process.
o) Integrated: Data from various sources are integrated
o) Time-variant: Trend analysis and long term reporting
o) non-volatile: Read only (not modified)
o) optimized for querying and Analysis

Architecture :

1) Source Systems : External data Sources (CRM, ERP)
2) ETL process (Extract, Transform, Load)
3) Data warehouse
4) Data marts
5) BI Tools



Advantages.

1) Centralized Data access
2) Improved Data Quality
3) Historical Analysis
4) Efficient query Performance
5) Support Decision-making

2. Data cube : Multi dimensional Data structure used in data warehousing and online analytical processing (OLAP)

operations :

* **Slice** : 3D -> 2D view of particular field
* **Dice** : Smaller Sub Cube
* **Roll up** : Aggregates data by climbing up a dimension hierarchy
* **Drill down** : Breaks down aggregated data into finer levels
* **Pivot** : Rotate the cube to view different perspective

3. Data Marts :

* Small Subdivision of Data warehouse

key features :

o) Subject oriented
o) Smaller Scope
o) optimized for specific use
o) derived from a Data warehouse
o) Improved performance

Types :

o) Dependent DM
o) Independent DM
o) Hybrid DM

Advantages

o) Faster
o) cost effective
o) de centralized control
o) Improved performance

4. **Data Lake**: It's like a real life lake, unstructured, elements of different types are there, raw, unstructured, flexible, schema-on-read model

Key features:

*) Raw Data storage
*) Scalable
*) Flexible schema
*) Cost effective
*) supports diverse data types
*) Accessible for Big Data Analytics

Advantages

1) Flexible
2) Scalability
3) Cost saving
4) Supports Advanced Analytic

Challenges:

1) Data Governance
2) Performance
3) Security

Advantages of Data Repo:

*) Centralized
*) Data Accessibilities
*) Cost efficiency
*) Enhanced security
*) Supports Advanced Analytics.

ETL (Extract, Transform, Load)

1) **Extract**: Getting the Data

Data Sources

1) Relational Databases, API's
2) FLAT files
3) NO SQL Data bases
4) Real-Time Data Streams
5) cloud Storage

Steps in Extraction:

1) Identify the data sources and their formats
2) Establish a connection to the sources
3) Extract the data efficiently, ensuring minimal disruption to the source system

challenges:

* Handling Large Data volumes
* Managing Incosistent formats
* Minimizing impact on source systems

2. Transform

1) Data cleaning
2) Data validation
3) Data Aggregation (grouping)
4) Data Maping (choose where the data can be stored)
5) Data enrichment (improving its accuracy)
6) Data formating (making it in proper format)
7) Data Hierarchy creation (Creates Hierarchies for drill down analysis)

3. Load:

Types:

* full load (Transforms all data)
* Incremental load (Transforms new or updated data)
* Real-Time load (Live)

Advantages of ETL

* Centralized data integration
* Improved data quality
* Historical analysis
* Enhanced decision making
* Scalability

ETL Tools:

* Informatic power centre
* Talend
* Aws glue
* Apache NIFY, (Data flow automation)
* google data cloud

Epilogue :

DA problems :

*) Refine the problem
*) Data collection
*) Data cleaning and preprocessing
*) Exploitry Data Analytics (Visualizing)
*) Data transformation
*) Data Modeling
*) Evaluation of results
*) Visualization and Reporting
*) Decision Making and action

Roles and Responsibilities of DA: —> roles " " -> responsibilities

*) Collection and "Integration of Data"
*) "Data cleaning and preprocessing "and transformation"
*) "EDA"
*) "Data Analysis and modeling"
*) "Reporting and communication", visualization
*) "Visualization"
*) "collaboration"
*) "Maintain data quality"

Skills required :

Tools and Tech

*) Data Base
*) Data cleaning libraries
*) Visualization tool
*) Big data tool

*) Technical skill

*) Data wrangling
*) EXEL
*) Data visualization
*) Statistical analytical tools SAS, SPSS
Statistical Analysis System
Statistical Package of Social Science
*) Machine learning

*) Soft skill

*) problem solving
*) communication
*) Attention to detail
*) collaboration

*) Business skill :

*) Domain *) critical thinking