Data Collection and Preprocessing Phase

Date	11 July 2024
Team ID	SWTID1720174920
Project Title	Human Resource Management: Predicting Employee Promotions Using Machine Learning
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	**Description**
Data Overview	Basic statistics, dimensions, and structure of the data.
Univariate Analysis	Exploration of individual variables (mean, median, mode, etc.).
Bivariate Analysis	Relationships between two variables (correlation, scatter plots).
Multivariate Analysis	Patterns and relationships involving multiple variables.
Outliers and Anomalies	Identification and treatment of outliers.
Data Preprocessing Code Screenshots	

Loading Data	Code to load the dataset into the preferred environment (e.g., Python, R).

Handling Missing Data	# Check for missing values
	df.isnull().sum()
	
	# Fill missing values in 'education' column with the mode
	df['education'] = df['education'].fillna(df['education'].mode()[0])
	print(df['education'].value_counts())
	
	# Fill missing values in 'previous_year_rating' column with the mode
	print(df['previous_year_rating'].value_counts())
	df['previous_year_rating'] = df['previous_year_rating'].fillna(df['previous_year_rating'].mode()[0])

Data Transformation	# Dropping unnecessary columns
	df = df.drop(['employee_id', 'gender', 'region', 'recruitment_channel'], axis=1)
	
	# Replacing missing values in 'education' and encoding it
	df['education'] = df['education'].replace(("Below Secondary", "Bachelor's", "Master's & above"), (1, 2, 3))
	
	# Label encoding for 'department'
	lb = LabelEncoder()
	df['department'] = lb.fit_transform(df['department'])
	
	# Handling outliers in 'length_of_service'
	q1 = np.quantile(df['length_of_service'], 0.25)
	q3 = np.quantile(df['length_of_service'], 0.75)
	IQR = q3 - q1
	upper_bound = q3 + 1.5 * IQR
	
	df['length_of_service'] = [upper_bound if x > upper_bound else x for x in df['length_of_service']]
	
	# Separating features and target variable
	x = df.drop('is_promoted', axis=1)
	y = df['is_promoted']

Feature Engineering	# Dropping unnecessary columns
	df = df.drop(['employee_id', 'gender', 'region', 'recruitment_channel'], axis=1)
	
	# Replacing missing values in 'education' and encoding it
	df['education'] = df['education'].replace(("Below Secondary", "Bachelor's", "Master's & above"), (1, 2, 3))
	
	# Label encoding for 'department'
	lb = LabelEncoder()
	df['department'] = lb.fit_transform(df['department'])
	
	# Handling outliers in 'length_of_service'
	q1 = np.quantile(df['length_of_service'], 0.25)
	q3 = np.quantile(df['length_of_service'], 0.75)
	IQR = q3 - q1
	upper_bound = q3 + 1.5 * IQR
	
	df['length_of_service'] = [upper_bound if x > upper_bound else x for x in df['length_of_service']]

Save Processed Data	# Assuming df as processed DataFrame
	df.to_csv('processed_emp_promotion.csv', index=False)