Dirty Data, Messy Data, Noisy Data, Inaccurate Data, Unreliable Data, Corrupted data, Garbage Data or Bad Data

Data which contains errors, inconsistencies, or other issues that make it challenging to work with or analyze effectively is known as dirty data.

Data Cleaning

Identifying that your data is not cleaned properly is crucial for ensuring the quality and reliability of your analyses and models. By paying attention to the following signs and conducting thorough data cleaning processes, you can improve the quality and reliability of your data-driven analyses and models.

Missing Values: Check for missing values in your dataset. Missing values can introduce bias and affect the accuracy of your analyses or models. Look for patterns in missing data and consider imputation techniques if appropriate.

Outliers: Outliers can skew statistical analyses and machine learning models. Visualize your data using box plots, histograms, or scatter plots to identify outliers. Consider whether these outliers are errors or genuine data points that require special treatment.

Inconsistent Formatting: Look for inconsistencies in the formatting of categorical variables or text fields. For example, variations in capitalization or spelling errors in category names can lead to difficulties in analysis.

Duplicate Records: Duplicate records can inflate counts and bias analyses. Use unique identifiers to identify and remove duplicate records from your dataset.

Data Distribution: Examine the distribution of your data to ensure it aligns with expectations. For example, if you're analyzing customer ages, check for unrealistic values or unusual spikes in certain age groups.

Data Validation: Perform data validation checks to ensure that values fall within expected ranges or categories. For instance, validate dates, numerical ranges, and categorical values to ensure they make sense in the context of your dataset.

Data Consistency: Check for inconsistencies between related variables. For example, if you have a dataset with sales records, ensure that the total sales for each month match the sum of individual sales records for that month.

Corrupted Data: Sometimes data can become corrupted during collection, storage, or transfer processes. Look for signs of corruption such as unreadable characters, incorrect file formats, or unexpected data types.

Unstructured Data: If you're working with unstructured data such as text or images, ensure that it's properly preprocessed and cleaned according to the requirements of your analysis or model.

Formatting, removing and imputing the data values are the only way to clean the data

Data imputation

Data imputation is a critical step in data analysis for handling missing values. Missing data can distort the results of analyses and lead to biased estimates if not properly addressed. Here are several general methods for data imputation commonly used in data analysis.

1. Mean/Median/Mode Imputation

Mean Imputation: Fill in the missing value with the mean of the non-missing values in the same variable.

Median Imputation: Use the median of the observed values. It's more robust to outliers compared to the mean.

Mode Imputation: For categorical data, missing values can be filled with the mode, or the most frequent category.

2. Random Imputation

Randomly select a value from the set of observed values for the variable and use it to fill in the missing value. This maintains the distribution but does not account for relationships between variables.

3. K-Nearest Neighbors (KNN) Imputation

Utilizes the K-nearest neighbors of the data point with missing values to impute data. It finds the 'k' observations closest to the missing data point and imputes using mean/mode from these neighbors.

4. Regression Imputation

Involves using a regression model to predict the missing value based on other variables. This method can maintain relationships among variables but might underestimate variability.

5. Multiple Imputation

A more sophisticated approach that fills in each missing value multiple times to create several complete datasets. Statistical analyses are performed on all datasets, and the results are pooled. It accounts for the uncertainty around the missing data.

6. Hot Deck Imputation

This method involves filling in a missing value with an observed response from a "similar" individual in your dataset. "Similarity" is defined based on certain criteria.

7. Interpolation and Extrapolation

Using statistical or machine learning techniques to estimate the missing values based on the trends/patterns found in the data. Time series data often benefits from these methods, especially if the data follows a predictable pattern over time.

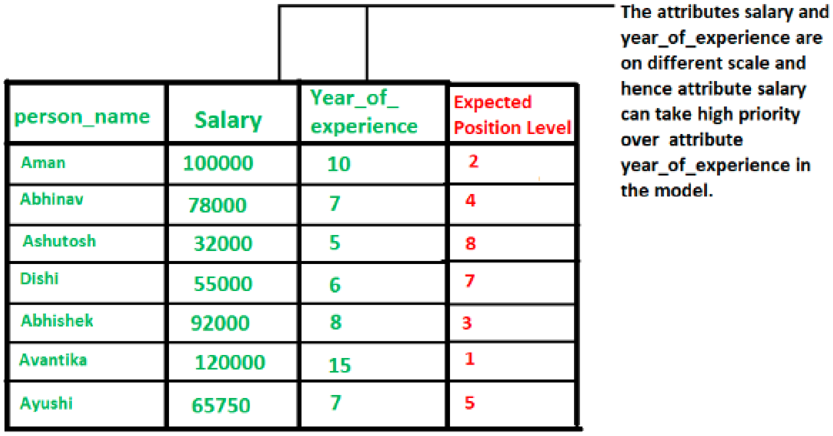
8. Advanced Machine Learning Methods

Deep Learning: Neural networks, such as autoencoders, can be trained to predict missing values based on the patterns learned from observed data.

Data conversion:

Dataset conversion involves changing the format or structure of a dataset from one type to another.

Data transformation: Involved the conversion of raw data into a more suitable format for analysis, modeling, or visualization.



person_name	Salary	Year_of_ experience	Expected Position Level
Aman	100000	10	2
Abhinav	78000	7	4
Ashutosh	32000	5	8
Dishi	55000	6	7
Abhishek	92000	8	3
Avantika	120000	15	1
Ayushi	65750	7	5

The attributes salary and year_of_experience are on different scale and hence attribute salary can take high priority over attribute year_of_experience in the model.

Normalization: To give all attributes an equal weight normalization is useful. Normalization is recommended for classification and clustering.

Min-max normalization

$$v_i' = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A.$$

z-score normalization

$$v_i' = \frac{v_i - \bar{A}}{\sigma_A},$$

Experiment 4

Objective:
Preparation for Dataset for analysis using Data Cleaning, Data imputation and Data conversion in R

a. **Cleaning**:
Removal of Null Values

```
head(airquality)
mean(airquality$Solar.R)
mean(airquality$Ozone)
mean(airquality$Wind)
mean(airquality$Solar.R, na.rm = TRUE)
mean(airquality$Ozone, na.rm = TRUE)
summary(airquality)
boxplot(airquality)
```

b. Imputation

```
New_df = airquality
New_df$Ozone = ifelse(is.na(New_df$Ozone), median(New_df$Ozone, na.rm = TRUE), New_df$Ozone)
summary(New_df)
boxplot(New_df)
```

c. Data Conversion: Transformation (transformation is a part of conversion)

Minmax

```
data <- c(10, 20, 30, 40, 50)
# Min-Max scaling function
min_max_scale <- function(x) {  (x - min(x)) / (max(x) - min(x))}
# Apply Min-Max scaling
scaled_data <- min_max_scale(data)
# Print the scaled data
print(scaled_data)
```

Zscore

```
data <- c(10, 20, 30, 40, 50)
# Z-score standardization function
z_score_scale <- function(x) {
(x - mean(x)) / sd(x)
}

# Apply Z-score standardization
```

```
standardized_data <- z_score_scale(data)
# Print the standardized data
print(standardized_data)
```

With datasets:

```
# Load a dataset (e.g., iris dataset)
data(iris)
# View the first few rows of the dataset
head(iris)
# Min-Max scaling function
min_max_scaling <- function(x)
{
 (x - min(x)) / (max(x) - min(x))
}
# Apply Min-Max scaling to the numeric columns of the dataset
iris_scaled <- as.data.frame(lapply(iris[, 1:4], min_max_scaling))
# Add column names to the scaled data
colnames(iris_scaled) <- paste0(colnames(iris[, 1:4]), "_scaled")
# View the first few rows of the scaled dataset
head(iris_scaled)
```