



**SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**
(Autonomous College Affiliated to the University of Mumbai)
NAAC ACCREDITED with "A" GRADE (CGPA : 3.18)



COURSE NAME: Machine Learning

CLASS: Third Year BTech

NAME: Shashwat Shah

BATCH: C22

EXPERIMENT NO. 1

AIM / OBJECTIVE:

To perform data preprocessing in terms of handling, missing data, removing outliers, eliminating duplicate rows and modifying the datatype, etc.

DESCRIPTION OF EXPERIMENT:

Python is an easy-to-learn programming language, which makes it the most preferred choice for beginners in Data Science, Data Analytics, and Machine Learning. It also has a great community of online learners and excellent data-centric libraries. With so much data being generated, it becomes important that the data we use for Data Science applications like Machine Learning and Predictive Modeling is clean. But what do we mean by clean data? And what makes data dirty in the first place? Dirty data simply means data that is erroneous. Duplicacy of records, incomplete or outdated data, and improper parsing can make data dirty. This data needs to be cleaned. Data cleaning (or data cleansing) refers to the process of “cleaning” this dirty data, by identifying errors in the data and then rectifying them. Data cleaning is an important step in and Machine Learning project, and we will cover some basic data cleaning techniques (in Python).

Cleaning Data in Python

We will now separate the numeric columns from the categorical columns.

Missing values

We will start by calculating the percentage of values missing in each column, and then storing this information in a DataFrame.

Drop observations

One way could be to drop those observations that contain any null value in them for any of the columns. This will work when the percentage of missing values in each column is very less.

Remove columns (features)

Another way to tackle missing values in a dataset would be to drop those columns or features that have a significant percentage of values missing.

Impute missing values



**SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**
(Autonomous College Affiliated to the University of Mumbai)
NAAC ACCREDITED with "A" GRADE (CGPA : 3.18)



There is still missing data left in our dataset. We will now impute the missing values in each numerical column with the median value of that column.

Outliers

An outlier is an unusual observation that lies away from the majority of the data. Outliers can affect the performance of a Machine Learning model significantly.

Duplicate records

Data can sometimes contain duplicate values. It is important to remove duplicate records from your dataset before you proceed with any Machine Learning project. In our data, since the ID column is a unique identifier, we will drop duplicate records by considering all but the ID column.

Fixing data type

Often in the dataset, values are not stored in the correct data type. This can create a problem in later stages, and we may not get the desired output or may get errors while execution.

PROCEDURE:

Describe the procedure that is used to carry out the experiment step-by-step. Describe every line of code with the proper interpretation of the output.

Perform data preprocessing with respect to your case study and discuss results of all the steps.

Code and output:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

df = pd.read_csv('/content/dirtydata.csv')
print(df.head(10))
```



**SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**
(Autonomous College Affiliated to the University of Mumbai)
NAAC ACCREDITED with "A" GRADE (CGPA : 3.18)



	Date	GameID	Drive	qtr	down	time	TimeUnder	TimeSecs	\
0	2009-09-10	2009091000	1	1	NaN	15:00	15	3600.0	
1	2009-09-10	2009091000	1	1	1.0	14:53	15	3593.0	
2	2009-09-10	2009091000	1	1	2.0	14:16	15	3556.0	
3	2009-09-10	2009091000	1	1	3.0	13:35	14	3515.0	
4	2009-09-10	2009091000	1	1	4.0	13:27	14	3507.0	
5	2009-09-10	2009091000	2	1	1.0	13:16	14	3496.0	
6	2009-09-10	2009091000	2	1	2.0	12:40	13	3460.0	
7	2009-09-10	2009091000	2	1	3.0	12:11	13	3431.0	
8	2009-09-10	2009091000	2	1	4.0	11:34	12	3394.0	
9	2009-09-10	2009091000	3	1	1.0	11:24	12	3384.0	
	PlayTimeDiff	SideofField	...	yacEPA	Home_WP_pre	Away_WP_pre	\		
0	0.0	TEN	...	NaN	0.485675	0.514325			
1	7.0	PIT	...	1.146076	0.546433	0.453567			
2	37.0	PIT	...	NaN	0.551088	0.448912			
3	41.0	PIT	...	-5.031425	0.510793	0.489207			
4	8.0	PIT	...	NaN	0.461217	0.538783			
5	11.0	TEN	...	NaN	0.558929	0.441071			
6	36.0	TEN	...	0.163935	0.578453	0.421547			
7	29.0	TEN	...	NaN	0.582881	0.417119			
8	37.0	TEN	...	NaN	0.617544	0.382456			
9	10.0	TEN	...	0.541602	0.591489	0.408511			
	Home_WP_post	Away_WP_post	Win_Prob	WPA	airWPA	yacWPA	Season		
0	0.546433	0.453567	0.485675	0.060758	NaN	NaN	2009.0		
1	0.551088	0.448912	0.546433	0.004655	-0.032244	0.036899	2009.0		
2	0.510793	0.489207	0.551088	-0.040295	NaN	NaN	2009.0		
3	0.461217	0.538783	0.510793	-0.049576	0.106663	-0.156239	2009.0		
4	0.558929	0.441071	0.461217	0.097712	NaN	NaN	2009.0		
5	0.578453	0.421547	0.441071	-0.019524	NaN	NaN	2009.0		
6	0.582881	0.417119	0.421547	-0.004427	-0.010456	0.006029	2009.0		

```

+ Code + Text
0 0.546433 0.453567 0.485675 0.060758 NaN NaN 2009.0
1 0.551088 0.448912 0.546433 0.004655 -0.032244 0.036899 2009.0
2 0.510793 0.489207 0.551088 -0.040295 NaN NaN 2009.0
3 0.461217 0.538783 0.510793 -0.049576 0.106663 -0.156239 2009.0
4 0.558929 0.441071 0.461217 0.097712 NaN NaN 2009.0
5 0.578453 0.421547 0.441071 -0.019524 NaN NaN 2009.0
6 0.582881 0.417119 0.421547 -0.004427 -0.010456 0.006029 2009.0
7 0.617544 0.382456 0.417119 0.034663 NaN NaN 2009.0
8 0.591489 0.408511 0.382456 0.026054 NaN NaN 2009.0
9 0.585485 0.414525 0.591489 -0.006884 -0.024520 0.018442 2009.0

[10 rows x 102 columns]
<ipython-input-1-bd9d4afaa28>:5: DtypeWarning: Columns (51) have mixed types. Specify dtype option on import or set low_memory=False.
df = pd.read_csv('/content/dirtydata.csv')

print(df.columns)
print(df.info())
print(df.describe())

Index(['Date', 'GameID', 'Drive', 'qtr', 'down', 'time', 'TimeUnder',
       'TimeSecs', 'PlayTimeDiff', 'SideofField',
       ...,
       'yacEPA', 'Home_WP_pre', 'Away_WP_pre', 'Home_WP_post', 'Away_WP_post',
       'Win_Prob', 'WPA', 'airWPA', 'yacWPA', 'Season'],
      dtype='object', length=102)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 43640 entries, 0 to 43639
Columns: 102 entries, Date to Season
dtypes: float64(35), int64(30), object(37)
memory usage: 14.0+ MB
None

```



**SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**
(Autonomous College Affiliated to the University of Mumbai)
NAAC ACCREDITED with "A" GRADE (CGPA : 3.18)



[]	count	4.364000e+04	43640.000000	43640.000000	36950.000000	43640.000000
	mean	2.009144e+09	12.335060	2.560151	2.015264	7.334166
	std	1.778114e+05	7.106666	1.126115	1.009782	4.659195
	min	2.009091e+09	1.000000	1.000000	1.000000	0.000000
	25%	2.009101e+09	6.000000	2.000000	1.000000	3.000000
	50%	2.009111e+09	12.000000	2.000000	2.000000	7.000000
	75%	2.009121e+09	18.000000	4.000000	3.000000	11.000000
	max	2.010010e+09	32.000000	5.000000	4.000000	15.000000
	TimeSecs	PlayTimeDiff	yrdIn	yrdline100	ydstogo	\
	count	43607.000000	43574.000000	43551.000000	43551.000000	43640.000000
	mean	1708.349554	20.799904	28.381323	47.692292	7.196471
	std	1057.388873	16.910683	13.129457	25.187992	4.796411
	min	-893.000000	0.000000	1.000000	1.000000	0.000000
	25%	803.000000	5.000000	20.000000	30.000000	3.000000
	50%	1800.000000	17.000000	30.000000	49.000000	9.000000
	75%	2597.000000	38.000000	39.000000	69.000000	18.000000
	max	3600.000000	234.000000	50.000000	99.000000	36.000000
	...	yacEPA	Home_WP_pre	Away_WP_pre	Home_WP_post	\
	count	16595.000000	40762.000000	40762.000000	40584.000000	
	mean	-0.400900	0.534956	0.465486	0.535202	
	std	2.008798	0.289938	0.289991	0.292223	
	min	-14.000000	0.000000	0.000000	0.000000	
	25%	-0.957404	0.327666	0.221636	0.323830	
	50%	0.000000	0.530724	0.469555	0.533272	
	75%	0.479230	0.778942	0.672906	0.783487	
	max	9.059733	1.000000	1.000000	1.000000	
	Away_WP_post	Win_Prob	WPA	airWPA	yacWPA	\
	count	40584.000000	40755.000000	4.296800e+04	16597.000000	16569.000000
	mean	0.465212	0.505817	1.841291e-03	0.014292	-0.010349
	std	0.302373	0.301700	4.576177e-03	0.057550	0.056050

```
[ ] #Dropping duplicates
df.drop_duplicates(inplace = True)
```

```
[ ] # get the number of missing data points per column
missing_values_count = df.isnull().sum()

# look at the # of missing points in the first ten columns
missing_values_count[0:10]
```

```
Date          0.0
GameID        0.0
Drive         0.0
qtr           0.0
down          0.0
time          0.0
TimeUnder     0.0
TimeSecs      0.0
PlayTimeDiff  0.0
SideofField   0.0
dtype: float64
```

```
[ ] total_cells = np.product(df.shape)
total_missing = missing_values_count.sum()
```

```
# percent of data that is missing
(total_missing/total_cells) * 100
```

```
24.974658974497224
```



SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC ACCREDITED with "A" GRADE (CGPA : 3.18)



```
+ Code + Test + Connect +
24.97A058B734H7224

[ ] df.fillna(df.mean(), inplace=True)

C:\python-input-6-5f93M754zb1>1: FutureWarning: The default value of numeric_only in DataFrame.mean is deprecated. In a future version, it will default to False. In a
df.fillna(df.mean(), inplace=True)

[ ] df.dropna(inplace=True)
```

CONCLUSION:

So, we successfully implemented cleaning of data using python