

# DATA CLEANING TECHNIQUES

## 1. Creation of data frame (unclean data set)

```
data=data.frame(x1=c(1:4,99999,1,NA,1,1,NA),
x1=c(1:5,1,"NA",1,1,"NA"),
x1=c(letters[c(1:3)],"x x","x","y y y","x","a","a",NA),
x4= "",
x5=NA)
print(data)
```

	x1	x1.1	x1.2	x4	x5
1	1	1	a		NA
2	2	2	b		NA
3	3	3	c		NA
4	4	4	x x		NA
5	99999	5	x		NA
6	1	1	y y y		NA
7	NA	NA	x		NA
8	1	1	a		NA
9	1	1	a		NA
10	NA	NA	<NA>		NA

## 2. Modify the column names

```
colnames(data)=c("col1","col2","col3","col4","col5")
print(data)
```

	col1	col2	col3	col4	col5
1	1	1	a		NA
2	2	2	b		NA
3	3	3	c		NA
4	4	4	x x		NA
5	99999	5	x		NA
6	1	1	y y y		NA
7	NA	NA	x		NA
8	1	1	a		NA
9	1	1	a		NA
10	NA	NA	<NA>		NA

## 3. Replace blank with NA

```
data[data == ""]=NA
print(data)
```

	col1	col2	col3	col4	col5
1	1	1	a	<NA>	NA
2	2	2	b	<NA>	NA
3	3	3	c	<NA>	NA
4	4	4	x x	<NA>	NA
5	99999	5	x	<NA>	NA
6	1	1	y y y	<NA>	NA
7	NA	NA	x	<NA>	NA
8	1	1	a	<NA>	NA
9	1	1	a	<NA>	NA
10	NA	NA	<NA>	<NA>	NA

#### 4. Preliminary Analysis

```
ncol(data)
```

```
nrow(data)
```

```
colSums(is.na(data))
```

```
rowSums(is.na(data))
```

```
> ncol(data)
[1] 5
> nrow(data)
[1] 10
> colSums(is.na(data))
col1 col2 col3 col4 col5
  2    0    1   10   10
> rowSums(is.na(data))
[1] 2 2 2 2 2 2 3 2 2 4
```

#### 5. Drop empty columns

```
data=data[,colSums(is.na(data)) !=nrow(data)]
```

```
print(data)
```

	col1	col2	col3
1	1	1	a
2	2	2	b
3	3	3	c
4	4	4	x x
5	99999	5	x
6	1	1	y y y
7	NA	NA	x
8	1	1	a
9	1	1	a
10	NA	NA	<NA>

## 6. Drop empty rows

```
data=na.omit(data)
```

```
print(data)
```

	col1	col2	col3
1	1	1	a
2	2	2	b
3	3	3	c
4	4	4	x x
5	99999	5	x
6	1	1	y y y
8	1	1	a
9	1	1	a

## 7. Remove duplicate observation

```
data=unique(data)
```

```
print(data)
```

	col1	col2	col3
1	1	1	a
2	2	2	b
3	3	3	c
4	4	4	x x
5	99999	5	x
6	1	1	y y y

## 8. To fix the datatype issue

```
sapply(data,class)
```

```
data=type.convert(data,as.is=TRUE)
```

```
sapply(data,class)
```

```
sapply(data,class) #checking the data type
      col1      col2      col3
"numeric" "character" "character"
data=type.convert(data,as.is=TRUE) #type conversion
sapply(data,class)
      col1      col2      col3
"integer"  "integer" "character"
```

## 9. Outlier treatment

```
boxplot_stats=boxplot.stats(data$col1)
print(boxplot_stats)
print(boxplot_stats$stats[1])
print(boxplot_stats$stats[5])
```

```
$stats
[1] 1.0 1.0 2.5 4.0 4.0

$n
[1] 6

$conf
[1] 0.5649031 4.4350969

$out
[1] 99999

> print(boxplot_stats$stats[1])
[1] 1
> print(boxplot_stats$stats[5])
[1] 4
```

## 10. finding the lower limit (under outlier treatment)

```
lower_fence=boxplot_stats$stats[1] -1.5* IQR(data$col1)
print(lower_fence)
```

```
[1] -2.75
```

## 11. Find the upper limit

```
upper_fence=boxplot_stats$stats[5] + 1.5* IQR(data$col1)
print(upper_fence)
```

```
[1] 7.75
```

## 12. Removing values less than lower fence and greater than upper fence

```
data=data[data$col1 >=lower_fence & data$col1 <= upper_fence, ]
print(data)
```

	col1	col2	col3
1	1	1	a
2	2	2	b
3	3	3	c
4	4	4	x x
6	1	1	y y y

### 13. Removing the spaces in between the text data

```
data$col3=gsub(" ","",data$col3)
print(data)
```

	col1	col2	col3
1	1	1	a
2	2	2	b
3	3	3	c
4	4	4	xx
6	1	1	yyy