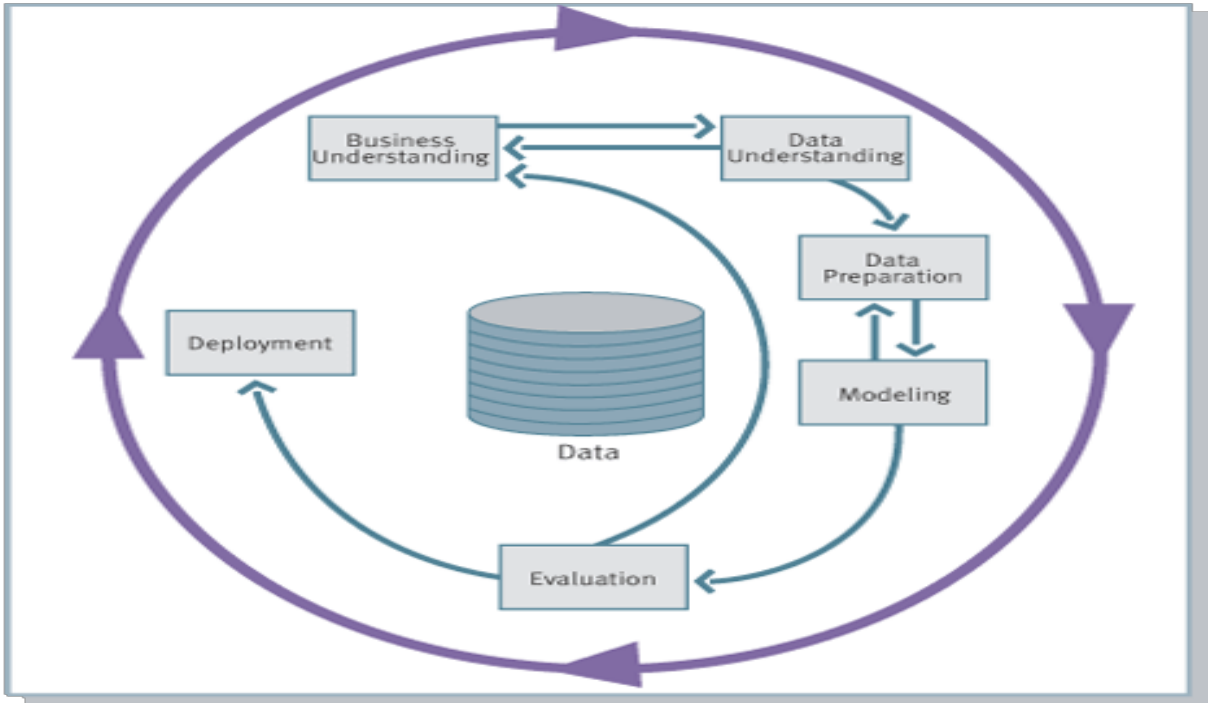
Crisp DM framework



Data Mining Tasks:
Description
Estimation
Prediction
Classification
Clustering
Association

Cross Industry Standard Process: CRISP-DM (*cont'd*)

(1) Business/Research Understanding Phase

- Define project requirements and objectives
- Translate objectives into data mining problem definition
- Prepare preliminary strategy to meet objectives

(2) Data Understanding Phase

- Collect data
- Perform exploratory data analysis (EDA)
- Assess data quality
- Optionally, select interesting subsets

(3) Data Preparation Phase

- Prepares for modeling in subsequent phases
- Select cases and variables appropriate for analysis
- Cleanse and prepare data so it is ready for modeling tools
- Perform transformation of certain variables, if needed

Cross Industry Standard Process: CRISP-DM (*cont'd*)

(4) Modeling Phase

- Select and apply one or more modeling techniques
- Calibrate model settings to optimize results
- If necessary, additional data preparation may be required for supporting a particular technique

(5) Evaluation Phase

- Evaluate one or more models for effectiveness
- Determine whether defined objectives achieved
- Establish whether some important facet of the problem has not been sufficiently accounted for
- Make decision regarding data mining results before deploying to field

Cross Industry Standard Process: CRISP-DM (*cont'd*)

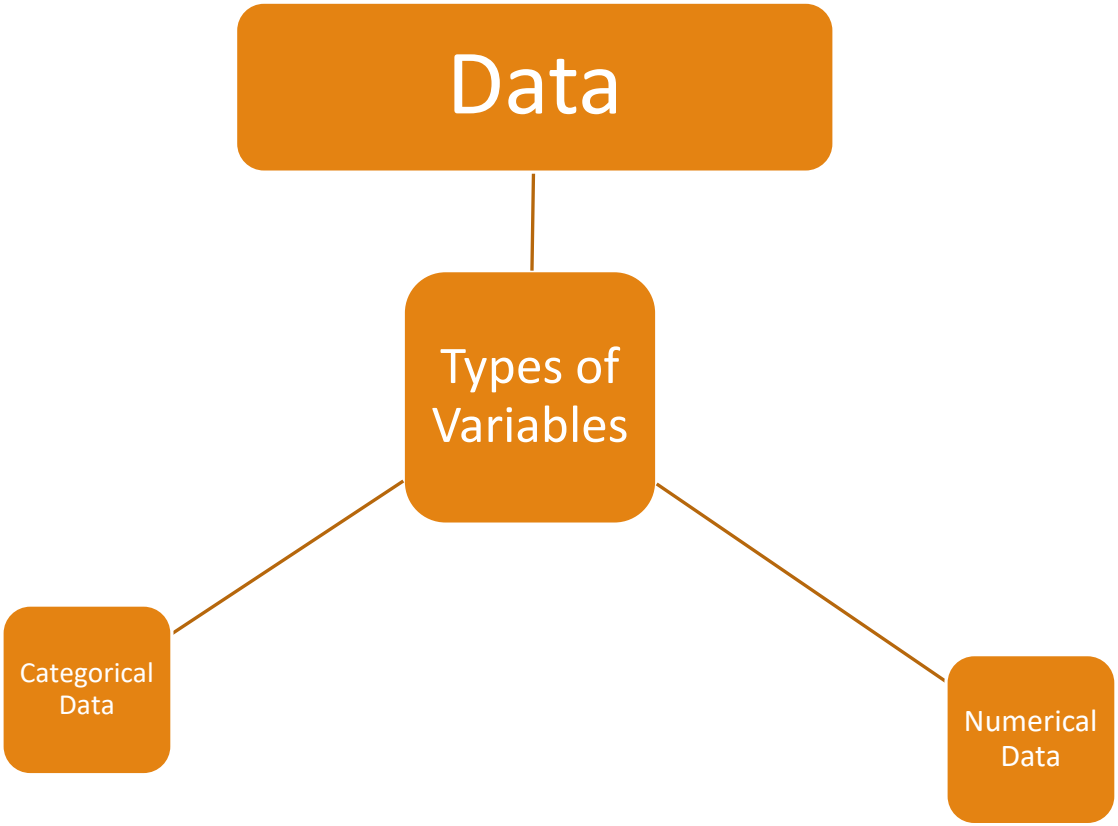
(6) Deployment Phase

- Make use of models created
- Simple deployment example: generate report
- Complex deployment example: implement parallel data mining effort in another department
- In businesses, customer often carries out deployment based on your model

Data Structure

No.	Gender	Age	Percentage SSC	Board SSC	Percentage HSC	Percentage Graduation Degree	Salary	Highest Degree (1: Graduation; 2: Post Graduation; 3= PhD)
1	M	23	62	Others	88	52	270000	1
2	M	21	76.33	ICSE	75.33	75.48	220000	2
3	M	22	72	Others	78	66.63	240000	3
4	M	22	60	CBSE	63	58	250000	1
5	M	22	61	CBSE	55	54	180000	2
6	M	23	55	ICSE	64	50	300000	2
7	F	24	70	Others	54	65	240000	1
8	M	22	68	ICSE	77	72.5	235000	3
9	M	24	82.8	CBSE	70.6	69.3	425000	3
10	F	23	59	CBSE	74	59	240000	1

Categorical (Qualitative Data) vs. Numerical Data (Quantitative Data)



Data Type

Cross-Sectional Data: A data collected on many variables of interest at the same time or duration of time is called cross-sectional data.

Time Series Data: A data collected for a single variable such as demand for smartphones collected over several time intervals (weekly, monthly, etc.) is called a time series data.

Panel Data: Data collected on several variables (multiple dimensions) over several time intervals is called panel data (also known as longitudinal data).

Why Do We Preprocess Data?

For data mining purposes, database values must undergo data cleaning and data transformation

Data often from legacy databases where values:

- Not looked at in years
- No longer relevant
- Missing

Minimize GIGO (Garbage In Garbage Out)

- IF garbage input minimized → THEN garbage in results minimized

Data preparation is 60%-80% of effort for data mining process

Example

Customer Id	Zip	Gender	Income	Age	Marital status	Transaction Amount
1001	10048	M	75000	C	M	5000
1002	J2S7K7	F	-40000	40	W	4000
1003	90210		10,000,000	45	S	7000
1004	6269	M	50,000	0	S	1000
1005	55101	F	99,999	30	D	3000
1006	55102	M	50,000	35	D	30000
1007	55102	M	50,000	35	D	35000

Churn Dataset

State: categorical, for the 50 states and the District of Columbia
Account length: integer-valued, how long account has been active
Area code: categorical
Phone number: essentially a surrogate for customer ID
International Plan: dichotomous categorical, yes or no
VoiceMail Plan: dichotomous categorical, yes or no
Number of voice mail messages: integer-valued
Total day minutes: continuous, minutes customer used service during the day
Total day calls: integer-valued
Total day charge: continuous, perhaps based on foregoing two variables
Total evening minutes: continuous, minutes customer used service during the evening
Total evening calls: integer-valued
Total evening charge: continuous, perhaps based on foregoing two variables
Total night minutes: continuous, minutes customer used service during the night
Total night calls: integer-valued
Total night charge: continuous, perhaps based on foregoing two variables
Total international minutes: continuous, minutes customer used service to make international calls
Total international calls: integer-valued
Total international charge: continuous, perhaps based on foregoing two variables
Number of calls to customer service: integer-valued
Total_Bill_Paid: integer-valued

Checking Data types

Any abnormalities Observed?

```
> str(data)
'data.frame':   3334 obs. of  23 variables:
 $ X               : int  1 2 3 4 5 6 7 8 9 10 ...
 $ State           : chr  "KS" "OH" "NJ" "OH" ...
 $ Account.Length  : int  128 107 137 84 75 118 121 147 117 141 ...
 $ Area.Code       : int  415 415 415 408 415 510 510 415 408 415 ...
 $ Phone           : chr  "382-4657" "371-7191" "358-1921" "375-9999" ...
 $ Int.l.Plan      : chr  "no" "no" "no" "yes" ...
 $ VMail.Plan      : chr  "yes" "yes" "no" "no" ...
 $ VMail.Message   : int  25 26 0 0 0 0 24 0 0 37 ...
 $ Day.Mins        : num  265 162 243 299 167 ...
 $ Day.Calls       : int  110 123 114 71 113 98 88 79 97 84 ...
 $ Day.Charge      : num  45.1 27.5 41.4 50.9 28.3 ...
 $ Eve.Mins        : num  197.4 195.5 121.2 61.9 148.3 ...
 $ Eve.Calls       : int  99 103 110 88 122 101 108 94 80 111 ...
 $ Eve.Charge      : num  16.78 16.62 10.3 5.26 12.61 ...
 $ Night.Mins      : num  245 254 163 197 187 ...
 $ Night.Calls     : int  91 103 104 89 121 118 118 96 90 97 ...
 $ Night.Charge    : num  11.01 11.45 7.32 8.86 8.41 ...
 $ Intl.Mins       : num  10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 8.7 11.2 ...
 $ Intl.Calls      : int  3 3 5 7 3 6 7 6 4 5 ...
 $ Intl.Charge     : num  2.7 3.7 3.29 1.78 2.73 1.7 2.03 1.92 2.35 3.02 ...
 $ CustServ.Calls  : int  1 1 0 2 3 0 3 0 1 0 ...
 $ Churn.          : chr  "False." "False." "False." "False." ...
 $ Total_Bill_paid: chr  "1,500" "2,300" "1,520" "2,320" ...
```

Range Constraint

Check for out of range value:

SAT score: 400-1600 ; Package weight: more than 0 lb/kg ; Adult heart rate: 60-100 beats per minute

Churn Data: Day.Charge > 0

```
Error: is_in_closed_range : data$Day.Charge are not all in the range [0,Inf].
There was 1 failure:
  Position Value   Cause
1       13 -21.9 too low
```

Day. Mins should be in the range 20-400

```
Error: is_in_closed_range : data$Day.Mins are not all in the range [20,400].
There were 9 failures:
  Position Value   Cause
1     1053  12.5 too low
2     1346     0 too low
3     1398     0 too low
4     1622  19.5 too low
5     1987   7.9 too low
6     2253  17.6 too low
7     2737   2.6 too low
8     2754   7.8 too low
9     3047  18.9 too low
```

Checking Membership for Categorical Data

Data	Example Values
Marital Status	Married, Unmarried
T-shirt Size	S, M, L, XL
Income_Category	< 20, 20-40, >40

Data (Churn)	Example Values
Vmail.Plan	Yes, No
Churn	True, False

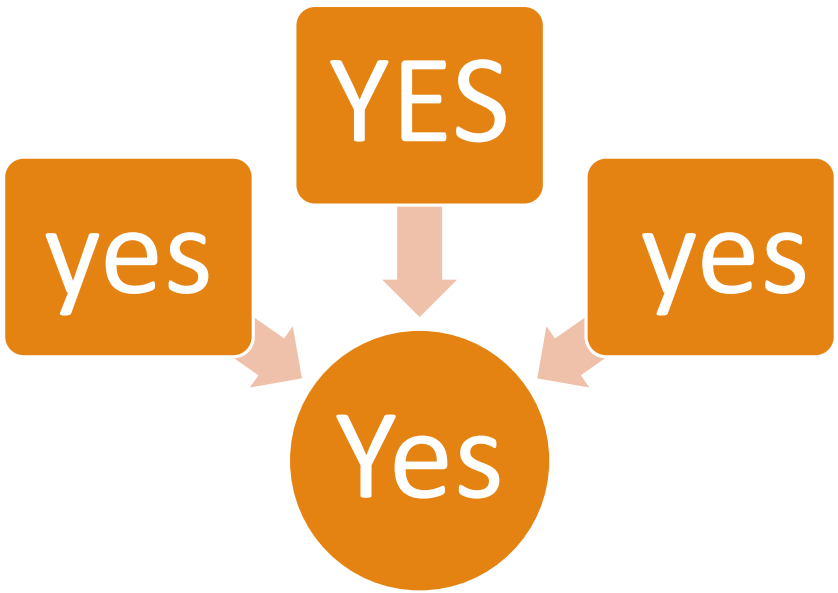
```
> levels(data$Churn.) [1] " False." "F" "False." "T" "True."
```

Checking Membership

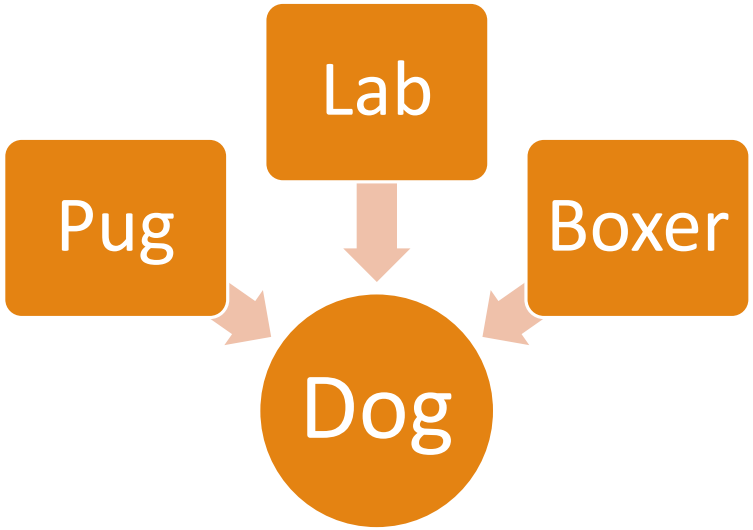
Inconsistency with a Category

Whitespace Inconsistency

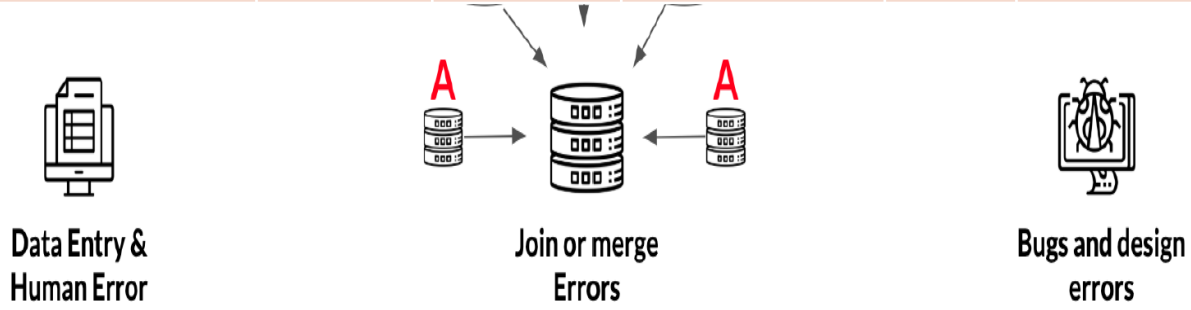
Case Inconsistency



Multiple Sub-Categories



Customer Id	Zip	Gender	Income	Age	Marital status	Transaction Amount
1001	10048	M	75000	C	M	5000
1002	J2S7K7	F	-40000	40	W	4000
1003	90210		10,000,000	45	S	7000
1004	6269	M	50,000	0	S	1000
1005	55101	F	99,999	30	D	3000
1006	55102	M	50,000	35	D	30000
1007	55102	M	50,000	35	D	35000



Full Duplicates

Partial Duplicates

```
> anyDuplicated(data2,fromLast = F) [1] 3334 > anyDuplicated(data2,fromLast = T) [1] 5
```

Uniformity

Different units or formats

Temperature: °C vs. °F

Weight: kg vs. g vs. lb

Money: USD $ vs. GBP £

Date: DD-MM-YYYY vs. MM-DD-YYYY vs. YYYY-MM-DD

Cross-Validation

Does this value make sense based on other values?

State	408	415	510
AK	14	24	14
AL	25	40	15
AR	13	27	15
AZ	15	36	13
CA	7	17	10
CO	25	29	12
CT	22	39	13
DC	14	27	13
DE	13	31	17
FL	12	31	20
GA	15	21	18
HI	15	30	8
IA	8	20	16
ID	12	41	20
IL	15	28	15
IN	18	33	20
KS	12	37	21
KY	15	32	12
LA	13	27	11
MA	24	29	12
MD	16	39	15
ME	15	25	22
MI	12	39	22
MN	20	40	24
MO	15	37	11
MS	15	31	19
MT	17	34	17
NC	25	28	15
ND	19	28	15
NE	13	34	14
NH	25	19	12
NJ	15	34	19
NM	16	35	11
NV	14	34	18
NY	19	47	17
OH	22	40	16
OK	17	27	17
OR	14	44	20
PA	14	19	12
RI	12	35	18
SC	13	30	17
SD	16	28	16
TN	11	30	12
TX	20	37	15
UT	12	37	23
VA	25	35	17
VT	17	36	20
WA	23	26	17
WI	22	35	21
WV	20	52	34
WY	17	41	19

- Each record has a value for Area Code and for State, there are only three distinct values for Area Code in the entire data set (408, 415, and 510)
- Each of the three values for Area Code is associated to each of the values for State.
- This presents an abnormality as each of these three Area Codes are for the state of California, so each record that does not have the State code CA has an invalid combination of State and Area Code values.

Missing Data

Dealing with missing data

Dealing with missing data

Simple approaches:

1. Drop missing data

2. Impute with statistical measures *(mean, median, mode..)* or domain knowledge

More complex approaches:

1. Impute using an algorithmic approach

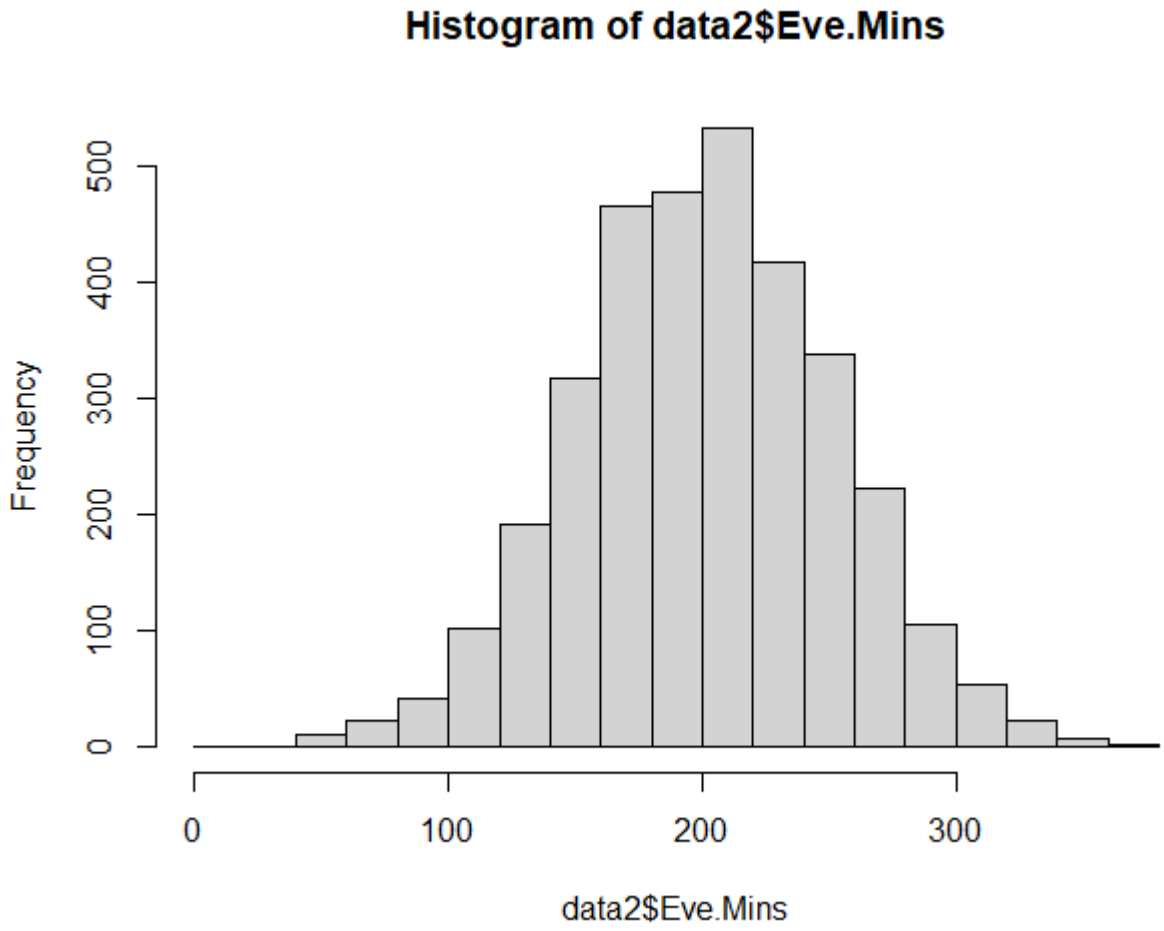
2. Impute with machine learning models

```
> sum(is.na(data2))
[1] 4
>
> # FInding the columns with missing values
> colSums(is.na(data2))
              X           State  Account.Length       Area.Code           Phone      Int.l.Plan
              0               0               0               0               0               0
     VMail.Plan  VMail.Message        Day.Mins       Day.Calls      Day.Charge        Eve.Mins
              0               0               0               0               0               2
      Eve.Calls      Eve.Charge      Night.Mins    Night.Calls    Night.Charge       Intl.Mins
              0               0               0               0               0               0
     Intl.Calls     Intl.Charge  CustServ.Calls          Churn. Total_Bill_paid
              0               0               2               0               0
>
```

Evening Minutes

Mean: 201.0

Median 201.3

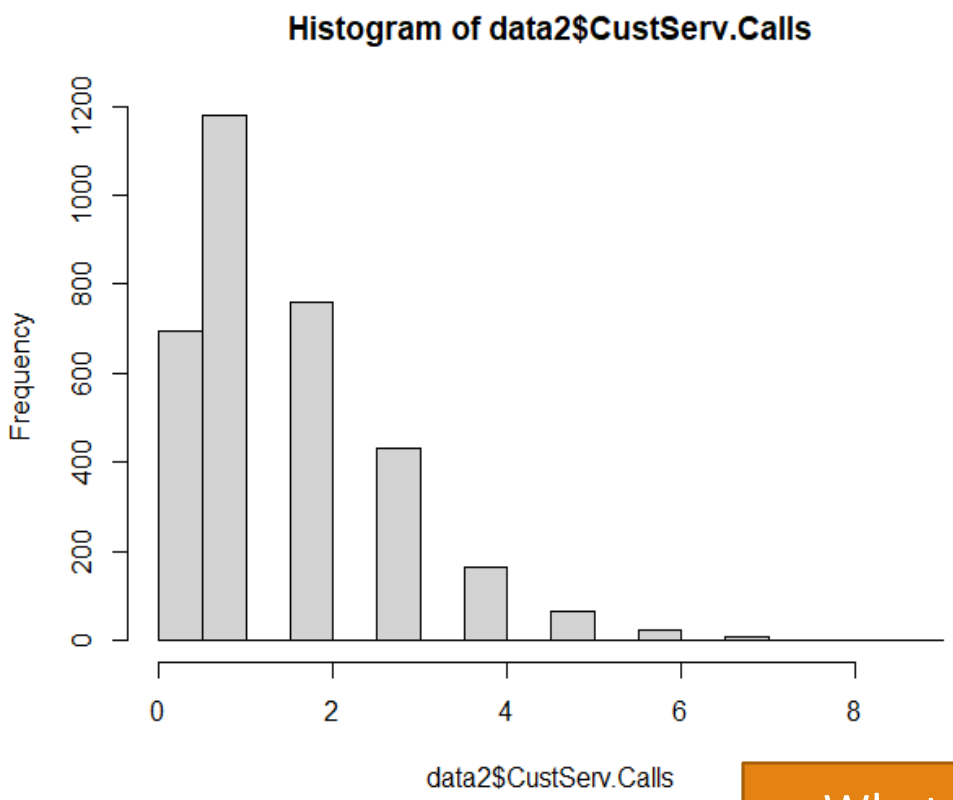


Replace missing values with mean

CustServ.Calls

Mean: 1.563

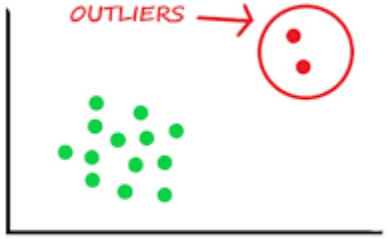
Median 1

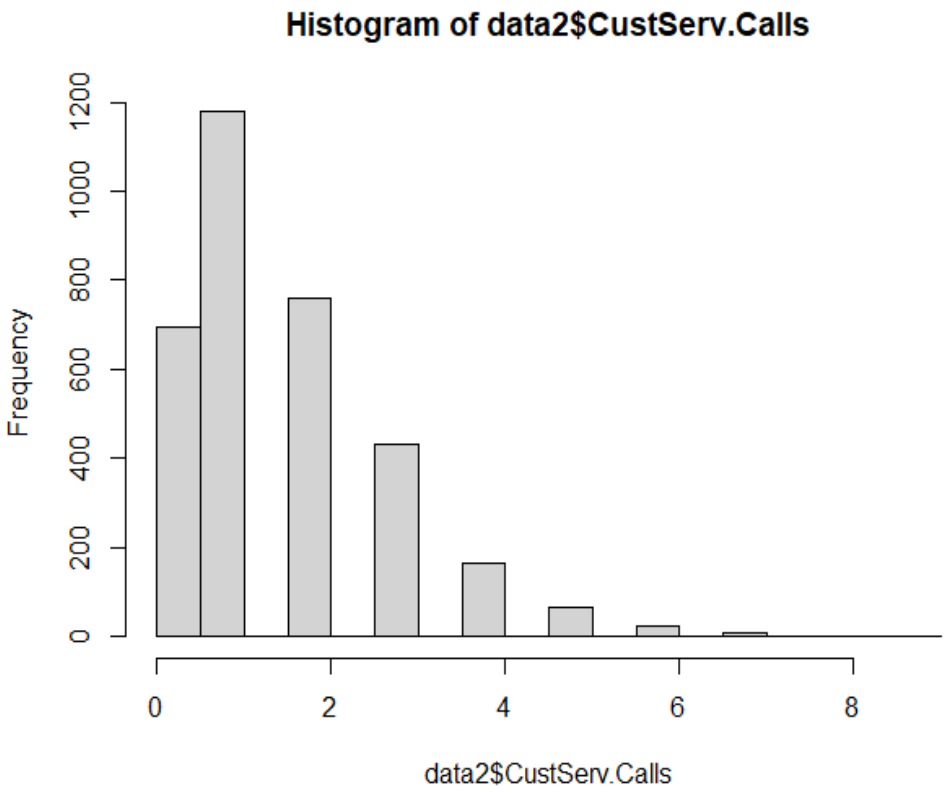


Replace missing values with median

What about missing values in VMail.Plan?

Outliers


OUTLIERS


Histogram of data2$CustServ.Calls
Frequency
0
200
400
600
800
1000
1200
0
2
4
6
8
data2$CustServ.Calls

Outliers

Outliers are values that lie near extreme limits of data range

Outliers may represent errors in data entry

Certain statistical methods very sensitive to outliers and may produce unstable results

Outliers detection

- Z- score method

  Calculate z-score for each point which is (x-mean(x))/sd(x)

  Any point greater than 3 or less than -3

- IQR method

  IQR <- Q3 –Q1

  Any point greater than Q3+1.5*IQR or less than Q1-1.5*IQR

The Z-scores for the number of service calls in descending order are as follows:

Z-Score(9) = (9 – 1.563) / 1.315 = 7.437 / 1.315 = **5.656** => Outlier

Z-Score(8) = (8 – 1.563) / 1.315 = 6.437 / 1.315 = **4.885** => Outlier

Z-Score(7) = (7 – 1.563) / 1.315 = 5.437 / 1.315 = **4.134** => Outlier

Z-Score(6) = (6 – 1.563) / 1.315 = 4.437 / 1.315 = **3.374** => Outlier

Z-Score(5) = (5 – 1.563) / 1.315 = 3.437 / 1.315 = 2.614 => Not an Outlier

Z-Score(4) = (4 – 1.563) / 1.315 = 2.437 / 1.315 = 1.853 => Not an Outlier

Z-Score(3) = (3 – 1.563) / 1.315 = 1.437 / 1.315 = 1.093 => Not an Outlier

Z-Score(2) = (2 – 1.563) / 1.315 = 0.685 / 1.315 = 0.521 => Not an Outlier

Z-Score(1) = (1 – 1.563) / 1.315 = -0.563 / 1.315 = -0.428 => Not an Outlier

Z-Score(0) = (0 – 1.563) / 1.315 = -1.563 / 1.315 = -1.189 => Not an Outlier

Q1 = 1

Q2 = 1

Q3 = 2

We then calculate IQR = Q3 – Q1 as follows:

IQR = 2 – 1 = 1.

Using the IQR, we calculate the upper and lower boundaries as follows:

Lower Bound = Q1 - 1.5IQR = 1 – 1.5(1) = -0.5

Upper Bound = Q3 + 1.5IQR = 2 + 1.5(1) = 3.5