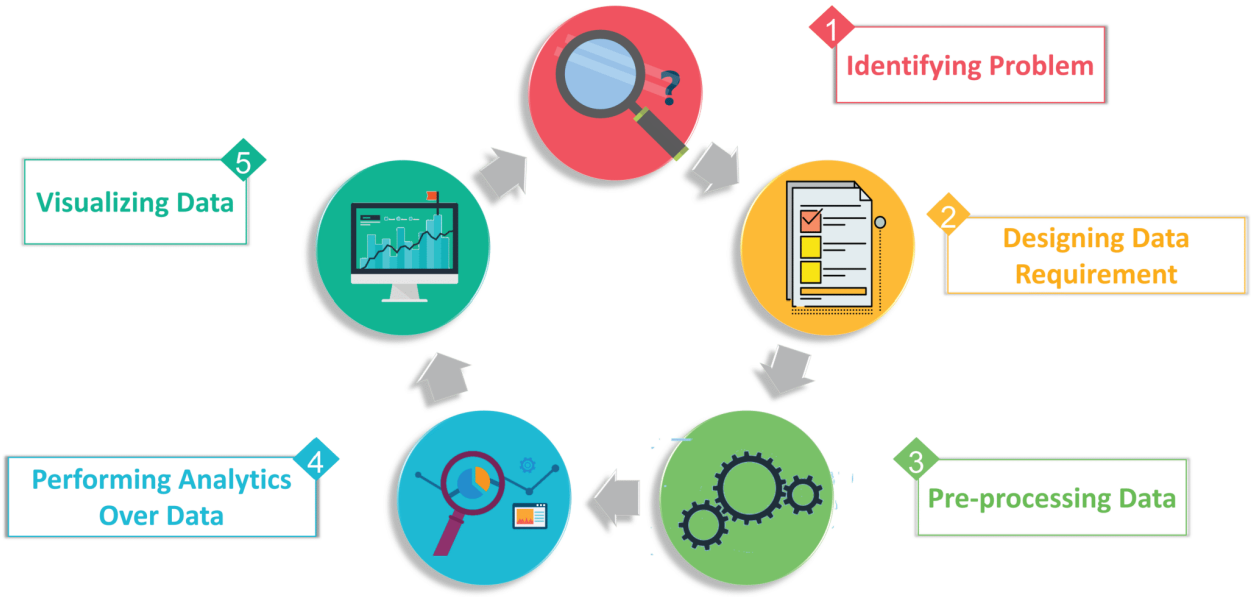
Data Preprocessing and Transformation

- Imdadullah Khan

Data Preprocessing



- Data preprocessing is a very important step
- It helps improve quality of data
- Makes it ready and more suitable for analytics
- Should be followed and guided by a thorough EDA
- EDA helps identify quality issues in data that are dealt with in this step

Issues with data

- Bad Formatting: Grade 'A' vs. 'a'
- Trailing Space: Extra spaces in commentary, white font ',' to avoid plagiarism detection
- Duplicates and Redundant Data: One ball repeated twice could be confused with a wide/No ball, one grade entered twice, could be confused with a course repetition
- Empty Rows: Could cause a lot of troubles during programming
- Synonyms, Abbreviations: rhb, right hand batsman, Fast Medium vs. Medium Pace
- Skewed Distribution and Outliers: Outliers could be points of interest or could be just noise, errors, extremities
- Missing Values: Missing grades, missing score
- Different norms and standards: miles vs. kilometers
- 1999: NASA lost equipment worth $125m because of an engineering mistake of not converting English to Metric unit

Steps in Preprocessing

Steps and subprocesses involved are performed when necessary

- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation

Data Cleaning

Also called data scrubbing, data munging, data wrangling

- Dealing with Missing values
- Noise Smoothing
- Correcting Inconsistencies
- Identifying Outliers

Data Cleaning: Missing Values

Missing data is very common and generally has significant consequences for analytics

- **Causes:**
  - Changes in experiments
  - human error, data entry error
  - measurement not possible
  - hardware failure
  - human bias
  - combining various datasets
- Missing values can have a meaning, e.g. absence of a medical test could mean that it was not conducted for a reason
- Knowing why and how data is missing could help in imputing

Data Cleaning: Missing Values

Knowing why and how data is missing could help in imputing

- **Missing Completely at Random (MCAR)**
  - Missingness independent of any observed or unobserved variables
- **Missing at Random (MAR)**
  - Missingness independent of missing values or unobserved variables
  - Missingness depend on observed variables with complete info
- **Missing Not at Ranodm (MNAR)**
  - Missingness depends on the missing values or unobserved variable

Data Cleaning: Missing Values - MCAR

Missing Completely at Random (MCAR)

- Missingness independent of any observed or unobserved variables
- Values of a variable being missing is completely unsystematic
- This assumption can somewhat be verified by examining complete and incomplete cases
- Data is likely representative sample and analysis unbiased

Age	25	26	29	30	30	31	44	46	48	51	52	54
IQ		121	91		110		118	93			116	

- Observe that values of age variable are roughly the "same" when IQ value is missing and when it is not

Data Cleaning: Missing Value - MAR

Missing at Random (MAR)

- Missingness independent of missing values or unobserved variables
- Missingness depend on observed variables with complete info
- The event that a value for Variable 1 is missing depends only on another observed variables with no missing values
- Not statistically verifiable (rely on subjective judgment)
- e.g. Only young people have missing values for IQ
- Shouldn't be that only high IQ people have missing values or
- Only males have IQ values missing (unobserved variable)

Age	25	26	29	30	30	31	44	46	48	51	52	54
IQ							118	93	116	141	97	104

Data Cleaning: Missing Value - MNAR

Missing Not at Ranodm (MNAR)

- Missingness depends on the missing values or unobserved variable(s)
- Pattern is non-random, non-ignorable, and typically arises due to the variable on which the data is missing
- Generally very hard to ascertain the assumption
- e.g. only low IQ people have missing values
- Or only males have missing IQ values

Age	25	26	29	30	30	31	44	46	48	51	52	54
IQ	133	121			110		118		116	141		104

Data Cleaning: Data Imputation

- Ignore the object (may lose many objects), what if class label is missing
- Manually fill in, works for small data and few missing values
- Use a global constant, e.g. MGMT Major, or Unknown, or ∞
- Substitute some measure of central tendency, e.g. mode, mean or median, or trimmed mean
- Missed Quiz: student mean, class mean, class mean in the quiz, class mean in quizzes, student mean in remaining quizzes, Duckworth-Lewis System, Mean of means, global mean
- Use class-wise mean or median, e.g. for missing players score in a match, use player's average, average of Pakistani's batsman, average of Pakistani batsmen against India, average of middle order Pakistani' batsmen again India in Summer in Sharjah
- Identify the most similar instance to this one (based on non-missing attributes values) and use value of that instance
- Can use top k (weighted by similarity) more on this later

Data Cleaning: Missing Value

There are several advanced techniques for missing values

- Expectation Maximization Imputation
- Regression based Imputation

Data Cleaning: Noise

- Random error or variance in measured data
- Elimination is generally difficult, Analytics should be robust so as quality is acceptable despite presence of noise
- Techniques to reduce effect of noise
- Smoothing by Binning:
  - Essentially replace each value by the average of values in the bin
  - Could be mean, median, midrange etc. of values in the bin
  - Could use equal width or equal depth (sized) bins
- Smoothing by local neighborhoods:
  - k-nearest neighbors, blurring, boundaries,
  - Smoothing is also used for data reduction and discretization
- Smoothing Time Series:
  - Moving Average
  - Divide by variance of each period/cycle

Data Cleaning: Correcting Inconsistencies

- Data can contain inconsistent values e.g. an address field with both ZIP code and city, but they don't match
- Some inconsistencies are easy to detect, e.g. negative age of a person
- Some inconsistencies may require consulting an external source
- Correcting an inconsistency requires additional or redundant information

Data Cleaning: Identifying Outliers

- Outliers are either
  - Data that have characteristics that are different from most of the other data (instance, observation, data item is an outlier)
  - Values of a variable that are unusual with respect to the typical values for that variable (a feature value is outlier)
- Unlike noise, outliers can be legitimate data or values
- Outliers could be points of interest

Data Integration and Data Combining

- Merging data from multiple sources
- RO and Admissions Data
- Cricinfo and PCB Data

Data Integration and Data Combining

- **Entity Identification Problem**
  - Sentiment Analysis on tweets about a cricket match to assess contribution of each of the 22 players
- Schema Integration
- Object Matching
  - Make sure that player ID in cricinfo dataset is the same as player code in PCB data (source of domestic games)
- Check metadata, names of attributes, range, data types and formats

Object Duplication

- Sometimes a whole instance/object etc. may be duplicated
- Occasionally two or more object can have all feature values identical, yet they could be different instances (e.g. two students with the same grades in all courses)

Data Integration and Data Combining

Redundancy and Correlation Analyses

- Redundant (not necessarily duplicate) features
- Sometimes caused by data integration (Data Duplication)
- If an attribute can be derived from one or more others, then it is redundant
  - e.g. if runs scored and balls faced are given, then no need to store strike rate
  - If aggregate score in course is given in absolute grading, then no need to store letter grade
- Covariance/Correlation and χ^2-statistics are used for pairs of numerical or ordinal/categorical attributes

Data Value Conflict Detection and Resolution

- Sometimes there are two conflicting values in different sources
- For example name is spelled differently in educational and NADRA's record
- This might need expert knowledge

Data Reduction

- Apart from duplicates removal etc.
- Because sometime we don't need all the data
- We reduce the data in either direction
- Reduce instances
- Reduce dimensions
- it helps a lot in reducing computational complexity and data visualization
- Can do sampling to reduce number of data objects i.e. obtain a representative sample of data

Data Reduction: Sampling

- **Equal probability sampling** of k out of n objects
- select objects from an ordered sampling window
- first select an object, then every $(n/k)th$ element (going circular)
- If there is some peculiar regularity in the how the objects are ordered, there is a risk of getting a very bad sample

Data Reduction: Sampling

- **Random Sampling** of k out of n objects
- Randomly permute objects (shuffle)
- Select the first k in this order
- Deals with the above regularity issue, but if there is big imbalance among classes or groups, we can get very bad sample

Data Reduction: Sampling

- **Stratified Sampling**
  - Suppose data is grouped into 3 groups (strata)
  - Randomly sample k/n fraction from each stratum
  - New sample will exhibits the distribution of population
  - Works for imbalanced classes but is computationally expensive

- **Clustered Sampling**
  - Cluster data items based on similarities (details later)
  - Randomly sample k/n fraction from each cluster
  - Efficient but not necessarily optimal, similarity definition is crucial
  - Underlying assumption is that similarity captures the classes

Data Reduction: Sampling

Imbalanced Classes: Classes or groups have huge difference in frequencies
The target class is rare

- Attrition prediction: 97% stay, 3% attrite (in a month)
- Medical diagnosis: 95% healthy, 5% diseased
- eCommerce: 99% do not buy, 1% buy
- Security: $>$ 99.99% of people are not terrorists
- Similar situation with multiple classes
- Predictions can be 97% correct, but useless
- Requires special sampling methods

Data Reduction: Feature Selection

- More importantly, one does dimensionality reduction
- For this we will study in quite detail the Curse of Dimensionality to see the problems associated with high dimensions and difficulties in dealing with higher dimensional vectors,
- We will discuss these techniques for dimensionality reduction (time permitting)
  - Locality Sensitive Hashing
  - Johnson-Lindenstrauss Transform
  - AMS Sketch
  - PCA and SVD

Data Reduction: Feature Selection

- Selecting a subset of the attributes to describe data objects
- All analysis is done based on selected attributes only
- Those features should be selected such that the probability distribution of class is roughly the same as the one obtained from all features
- This is a huge topic on its own
- It is a fascinating topic too, anyone interested in classification is encouraged to review the relevant literature

Data Transformation

- **Purpose**
  - Analytics is more efficient
  - Analytics is more meaningful
  - Visualization is more meaningful and easier
- **Steps:** include
- Ordinal to Numeric
- Smoothing
- Attribute Construction
- Aggregation (e.g. GPA from grades)
- Discretization and Quantization needed e.g. for decision tree, rule based systems
- Standardization, scaling and normalization
- More on some specific transformations later

Data Transformation

- Data transformations are applications of some mathematical modification to the values of a variable
- The goal is to make an entire set of values have a particular property (for example same range, same unit (or lack thereof))
- Variety of possibilities are there for different applications from adding constants (to shift the data to a manageable min e.g. shifting to positive)

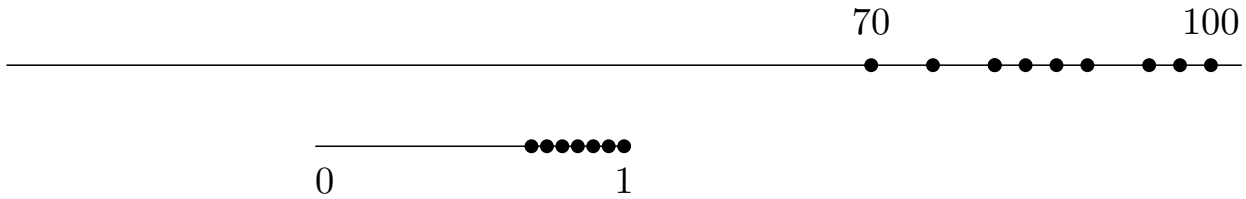
Standardization or Scaling

- scaling data so it falls in a smaller, comparable or manageable range
- data could be in different units e.g. kilometers and miles
- Units might not be known
- small units means larger values and larger ranges
- In values of "norms" and many distance measures, attributes of smaller units get more weights than attributes with larger units
- transform data to fall in small and importantly common range
- This way all attributes get the same weight, this has huge implications in distance/similarity computation and hence in applications like clustering and recommendation systems
- We will discuss this in more detail, if you remind me, when we talk about norms and distance measures

MAX-MIN Standardization/Scaling

Transform the data (an attribute X) to the interval $[0, 1]$

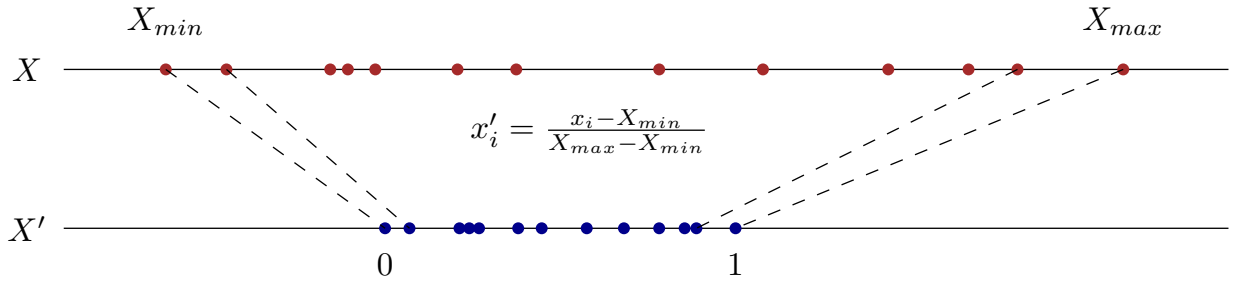
- Could do just $x_i' = x_i / X_{max}$, new max is 1
- Might get very narrow range within $[0, 1]$



MAX-MIN Standardization/Scaling

Transform the data (an attribute X) to the interval $[0, 1]$

- First shift everything to $[0, sth]$ by subtracting X_{min}
- $x'_i = \dfrac{x_i - X_{min}}{X_{max} - X_{min}}$



- Preserves relationships among original objects, max, min, median and all quantiles are the same objects
- If attribute Y is also scaled similarly, then X and Y are comparable
- We get different (scaled) std-dev, can suppress effect of outliers
- Two sections one with very harsh grading and other with extremely lenient grading, GIKI and LUMS GPA's
- Note that unit of X remains the same

z-score Normalization

- If we don't know min and max (don't have full data), or when outliers dominates, max-min scaled data is harder to interpret
- Transform the data to a scale with mean 0 and std-dev 1
- $x_i' = \dfrac{x_i - \overline{x}}{\sigma_x}$
- Stable data, common scale, all variables are unit-less and scalar
- Resulting vectors have properties of standard normal $\mu = 0, \sigma = 1$
- Again the relative order of points is maintained
- It makes no difference to the shape of a distribution

Sec1	90	10	50	30	40	80	74	68	61
Sec2	63	40	35	38	21	18	28	19	30
Sec1	1.4	-1.9	$-.24$	-1.07	$-.65$	.99	0.75	.5	.21
Sec2	2.3	.3	$-.14$	.13	.3	-1.6	$-.74$	.04	$-.57$

Other families of transformation

- In statistical analysis we often transform a variable x by a function $f(x)$ of that variable
- It changes the shape of the distribution of x or the relationship of x with some other variable y
- "Transformations are needed because there is no guarantee that the world works on the scales it happens to be measured on"
- Often it helps and is needed to transform the results back to the original scale by taking the inverse transform

Reasons for Transformation

- Convenience
- Reducing skew
- Equal Spreads
- Linear relationship
- Additive relations
- For one variable the first three reasons apply

Reasons for Transformation

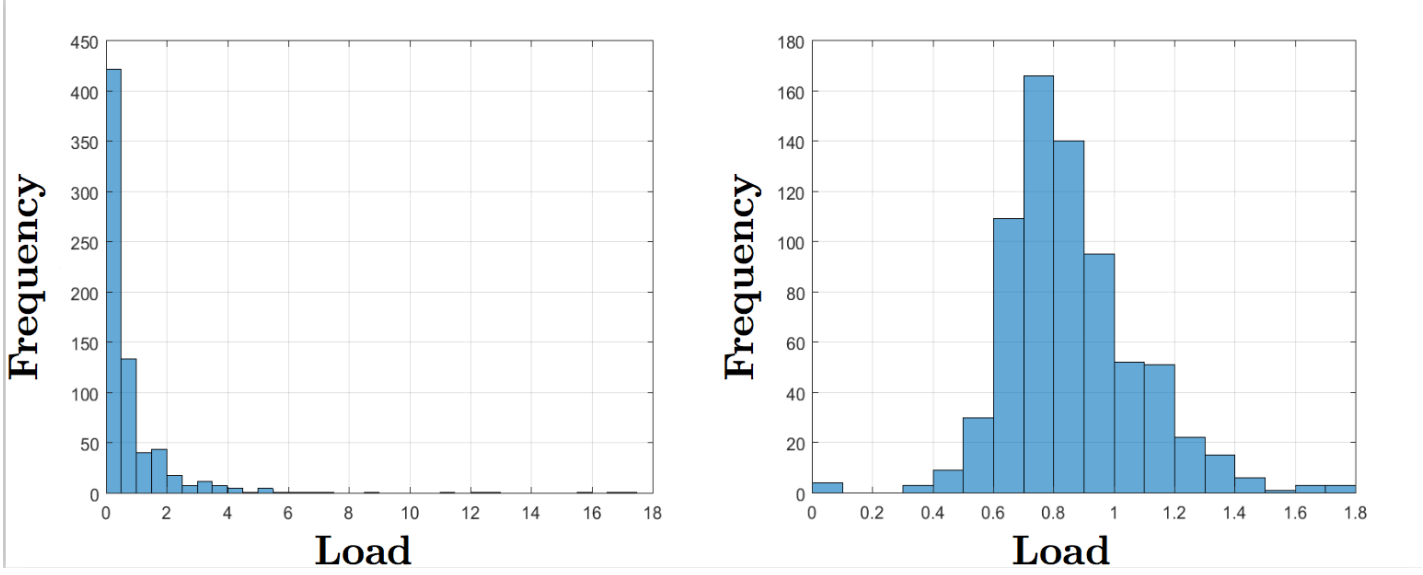
Convenience

- The transformed scale may be as natural as the original and more convenient for a specific purpose
- Since transformation often change units, one can transform the data to a unit that is easier to think about
- z-score normalization is extremely useful for comparing variables expressed in different units
- Rather than working with 101/120, 130/140, and 10/73, its easier to work with percentages. We might want to work with sines rather than degrees

Reasons for Transformation

Reducing Skewness

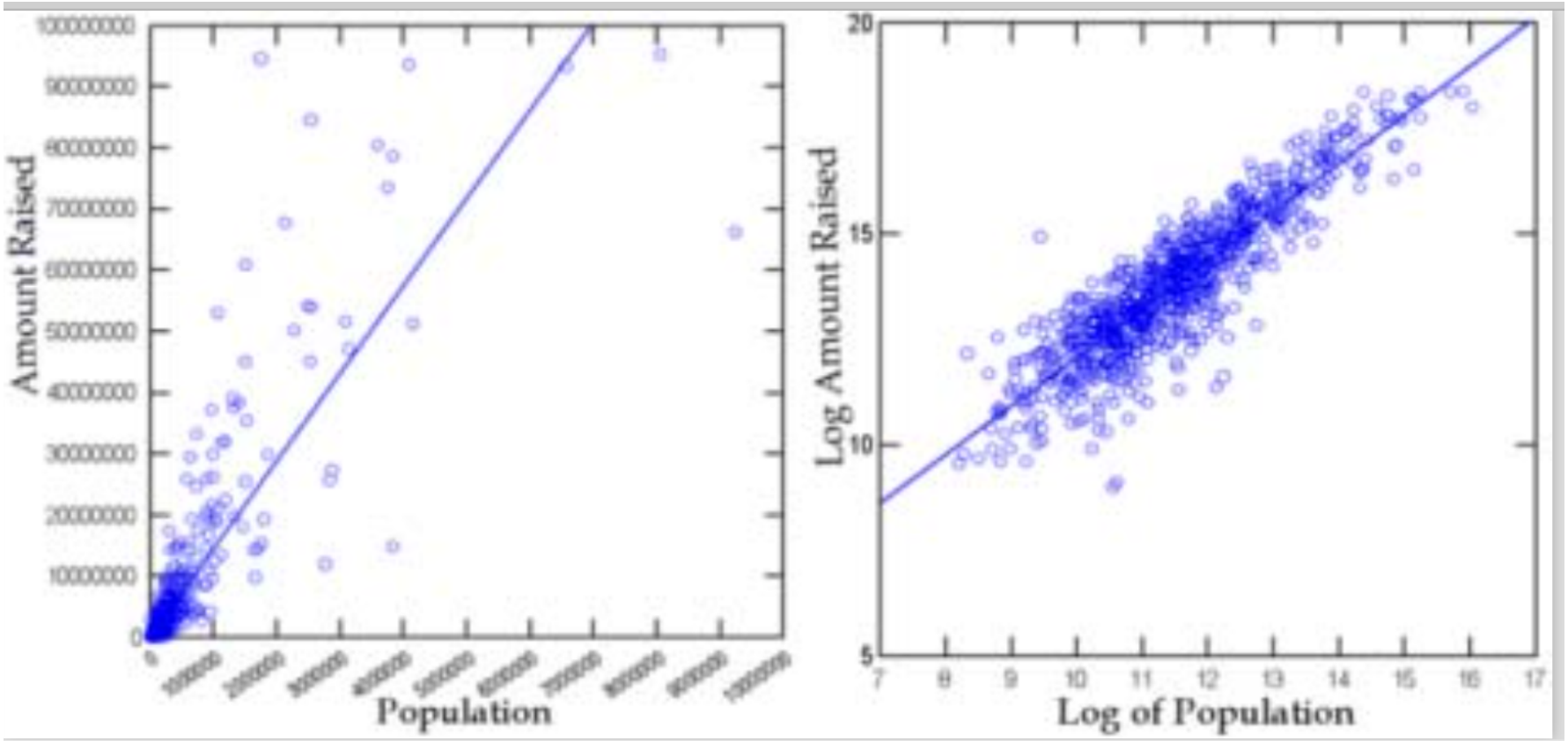
- Many parametric statistical model assume that data is coming from a certain distribution with fixed parameters
- Generally the assumption is of normality in data (easier)
- In order to say something like the probability of observing a particular max/mean etc. such an assumption is needed
- Assumption doesn't have to be true
- Data might have skew, (transform -taking root, log, or power)



Reasons for Transformation

Equal Spread, Homoskedasticity

- Sometime data is transformed to achieve approximately equally spread across the regression line (marginals)
- Subsets of dataset having roughly equal spread is a condition called **homoskedasticity**
- Its opposite property is **heteroskedasticity**



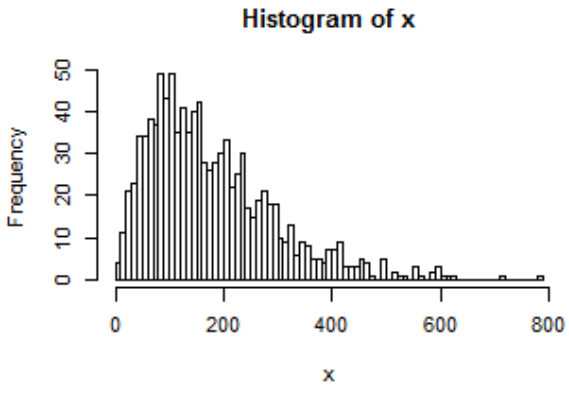
Common Transformations

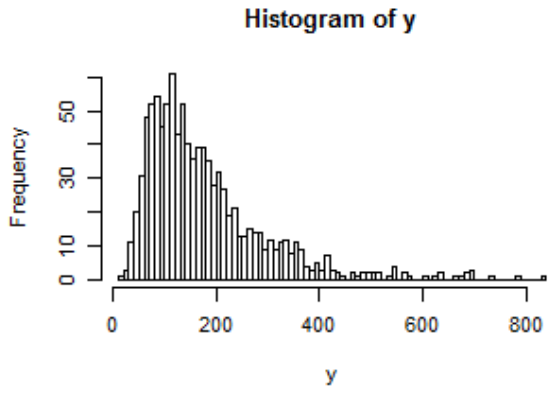
- All of the following transformations improves normality
- Some reduce the relative distance among values while still preserving the relative order
- Reduce the relative distance of values on the right sides (larger values) more than the values on the left side
- They are used to reduce right skew of data
- Issue of dealing left skew of data is discussed afterwards

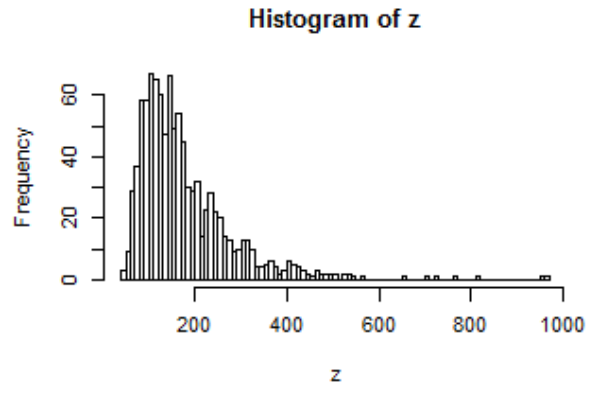
Logarithms

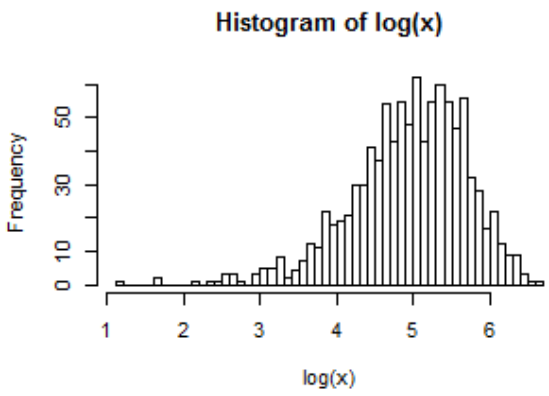
- $x' = \log x$
- major effect on the shape of the distribution
- Commonly used to reduce right skewness
- Often appropriate for measured variables (real numbers)
- Since log of negative numbers are not defined and that of numbers $0 < x < 1$ are negatives, we must shift values to a minimum of 1.00
- logs with different bases can be used (commonly used are natural logs, base 2 and base 10.) One often tries multiple first to settle on one
- Higher bases pull larger values drastically

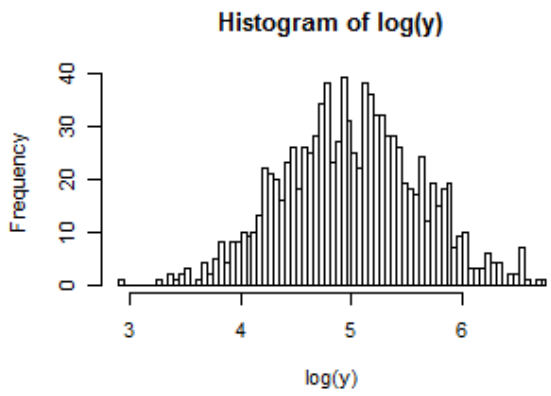
Logarithms

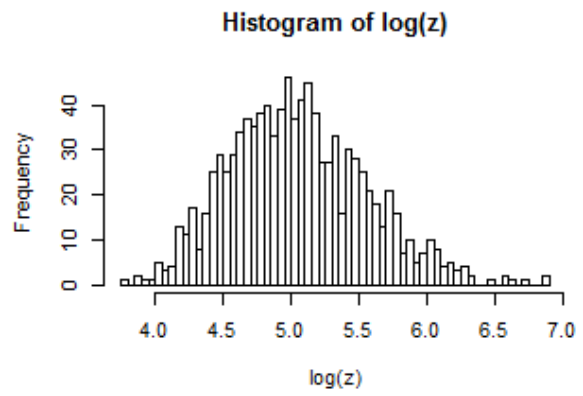

Histogram of x
Frequency
0
10
20
30
40
50
0
200
400
600
800
x


Histogram of y
Frequency
0
10
30
50
0
200
400
600
800
y


Histogram of z
Frequency
0
20
40
60
200
400
600
800
1000
z

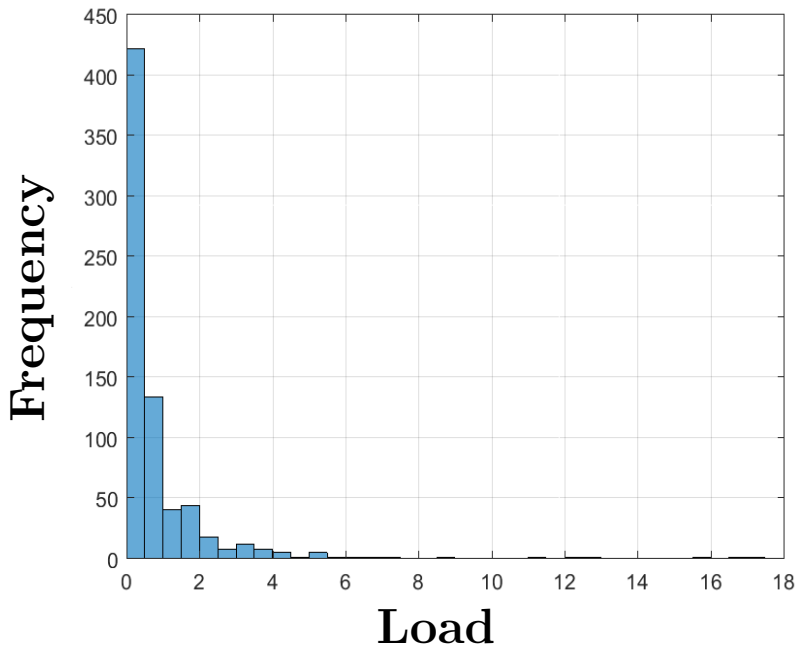

Histogram of log(x)
Frequency
0
10
30
50
1
2
3
4
5
6
log(x)

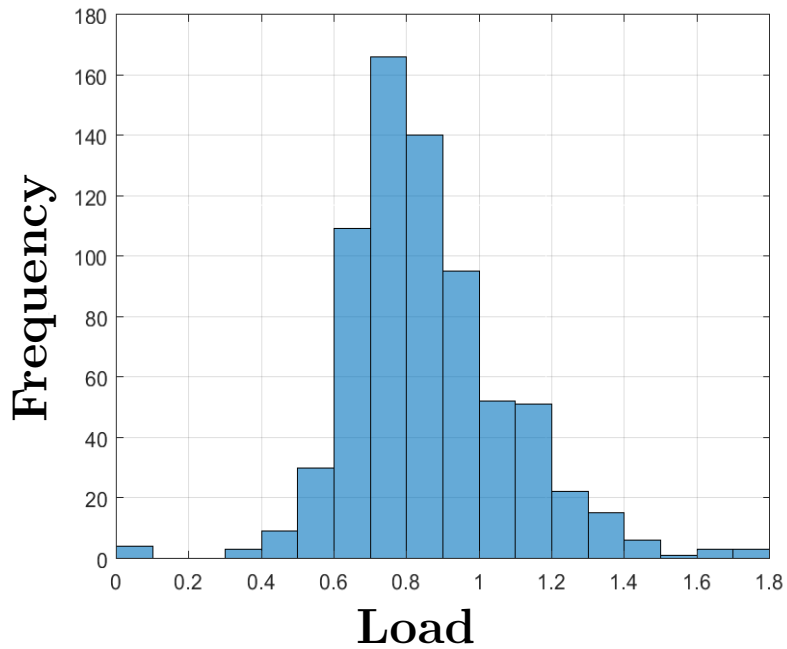

Histogram of log(y)
Frequency
0
10
20
30
40
3
4
5
6
log(y)


Histogram of log(z)
Frequency
0
10
20
30
40
4.0
4.5
5.0
5.5
6.0
6.5
7.0
log(z)

Cube Root

- $x' = x^{1/3}$
- significant effect on shape of distribution, weaker than log
- reduces right skew
- can be applied to 0 and negative numbers
- cube root of a volume has the units of a length





Square Root

- $x' = \sqrt{x}$
- Reduces right skew,
- square root of an area has unit of a length
- Commonly applied to counted data
- Negative values must first be shifted to positives
- Important consideration: roots of $x \in (0, 1)$ is $\geq x$, while roots of $x \in [1, \infty)]$ decreases ($\leq x$), so we must be careful
- Might not be desirable to treat some number differently than others, though the relative order of values will be maintained

Reciprocal and Negative Reciprocal

- $x' = 1/x$ or $x' = -1/x$
- Cannot be applied to 0
- Should be applied when all data is positive or negative
- population density (people per unit area) becomes area/person
- persons per doctor becomes doctors per person
- rates of erosion become time to erode a unit depth
- Reciprocal reverses order among values of the same sign
- It basically makes very large number very small and very small numbers very large
- Negative reciprocal preserves order among values of the same sign, this is commonly used
- This has the strongest effect

Left Skewed Data: Squares and higher powers

All the above transformation essentially deal with right skew, to deal with left skew one first reflects the data (multiply -1), and then apply these transformations. Generally one would need to shift the data to a new minimum of 1.0 after reflection and then apply the transform

- $x' = x^2$
- moderate affect on distribution shape
- can be used to reduce left skew

Transformation to make linear relationship

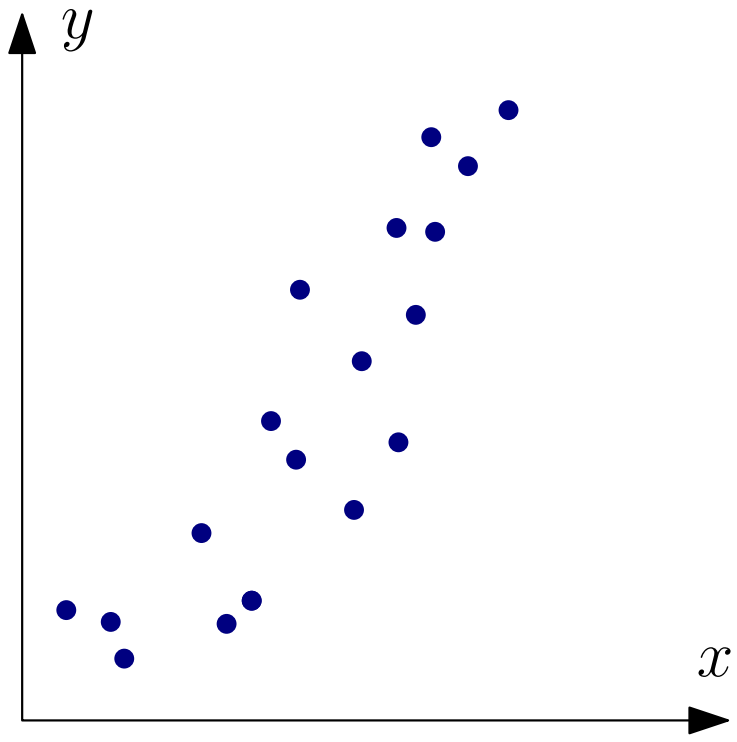
- Suppose we want to describe a variable y in terms of x
- We want to express it as linear relationship

$$y = ax + b$$

- Transformation in many cases helps us fit a good line

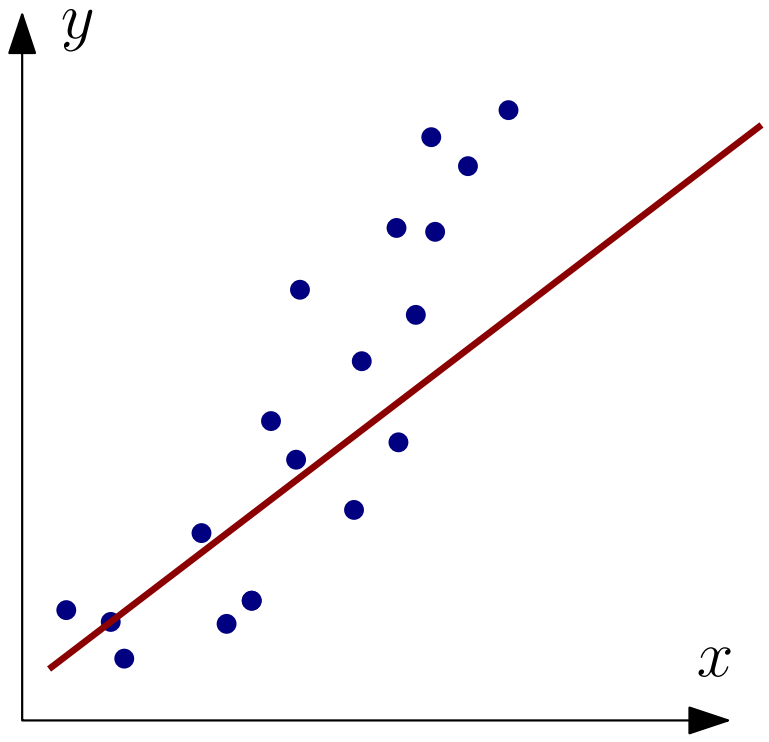
Transformation to make linear relationship

$$y = ax + b$$



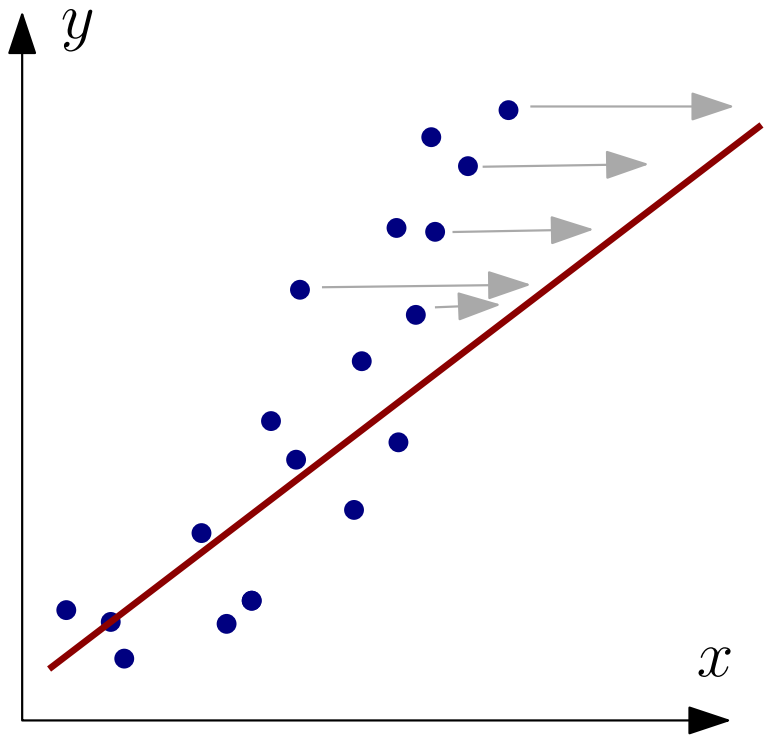
Transformation to make linear relationship

$$y = ax + b$$



Transformation to make linear relationship

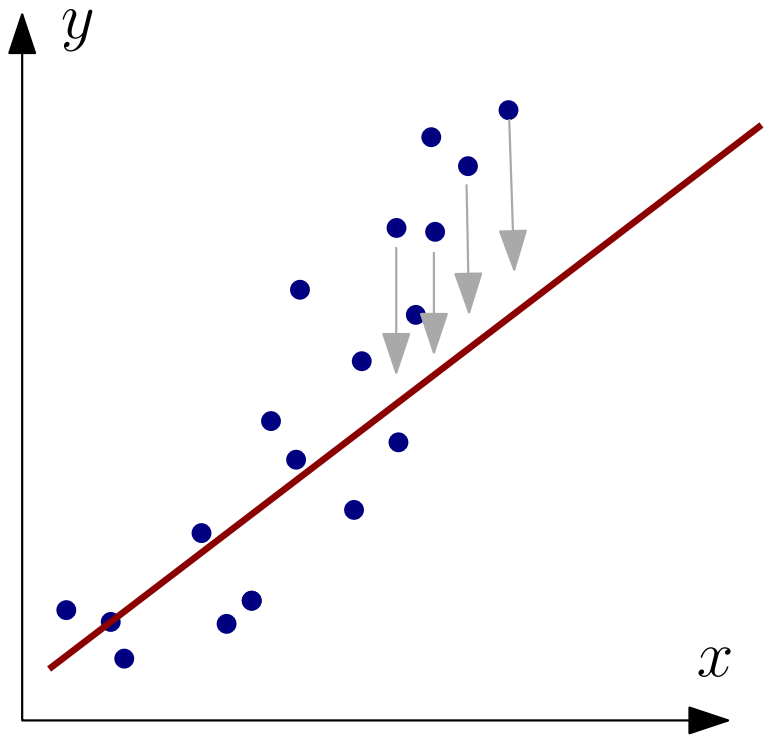
$$y = ax + b$$



Instead, express as $y = ax^2 + b$

Transformation to make linear relationship

$$y = ax + b$$



Can also do $\log y = ax + b$