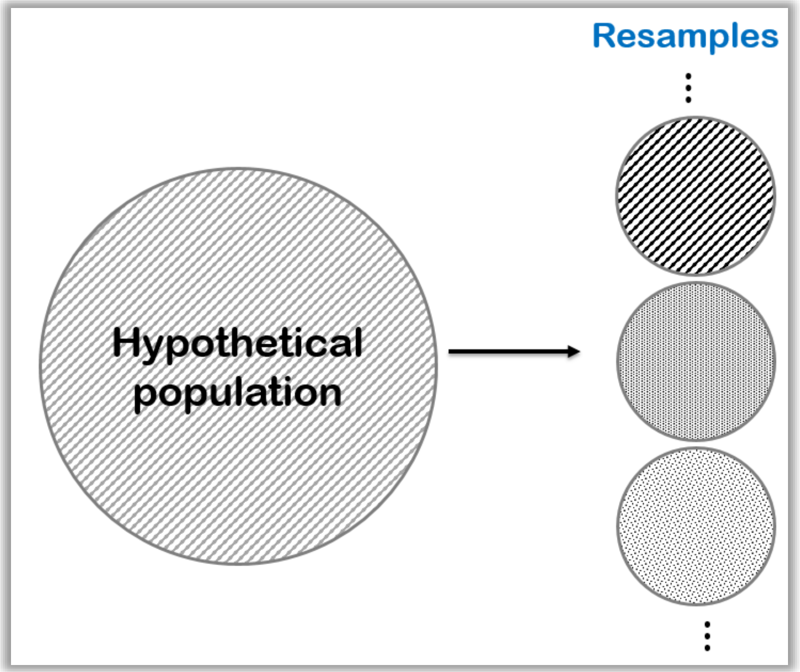
# Sampling: Data Pre-processing




## A resampling procedure; pros and cons of different sampling schemes; bootstrap sampling; sampling bias; sample size

*“Big data are not necessarily good data; well-designed small sample surveys can produce more accurate results than huge datasets that are just lying around.“*

**Prof. Nagiza F. Samatova**
samatova@csc.ncsu.edu
Department of Computer Science
North Carolina State University

NC STATE Executive Education

February 2019

# Learning Objectives: Sampling

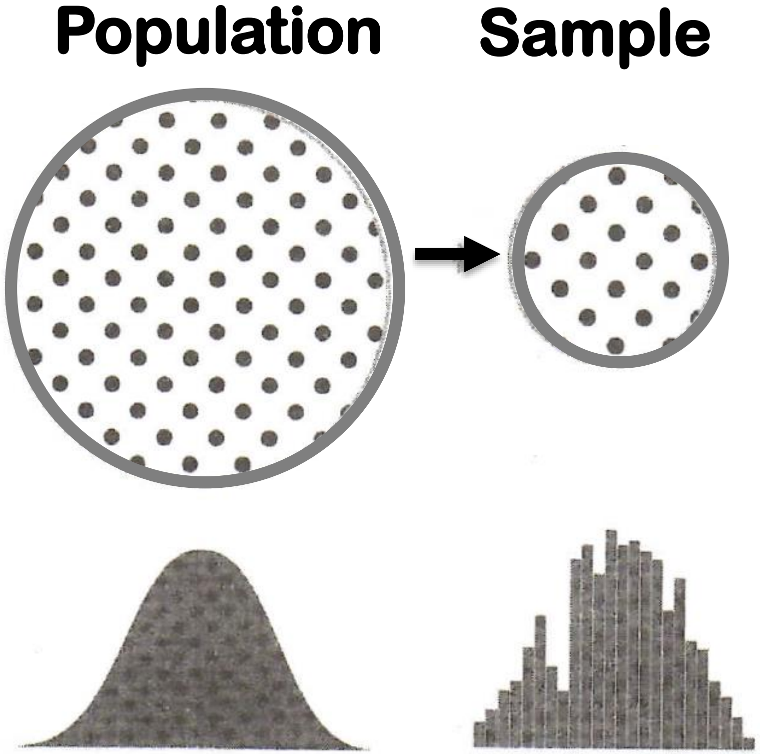
- **Specify what is required for a simple random sample (SRS)**
- **Specify the resampling procedure to determine:**
  - the sampling distribution of a proportion
  - the sampling distribution of a mean
- **Understand pros and cons of different statistical sampling schemes:**
  - random, stratified, cluster, self-selection
- **Understand and use bootstrap and permutation sampling**
- **Understand the meaning of glossary terms:**
  - populations, samples, parameters, statistic, sampling frame, bias (see Glossary)
- **Understand sampling procedures:**
  - Explain the relationship between required sample size for different population sizes
  - Explain bias caused by self-selection and non-response in surveys

# Sampling: Basic Terminology

| Term | Definition | Examples/Comments |
|---|---|---|
| **Parameter** | A measurable characteristic of the population | mean, proportion |
| **Population** | The target group of study | California voters (eligible to vote? vs. registered?) |
| **Sample** | A subset of the population. If drawn randomly, then it is a random sample | |
| **Sampling frame** | A practical representation of the population | Only registered voters |
| **Statistic** | A measurable characteristic of a sample used to estimate a population parameter | empirical mean is a statistic for a theoretical mean |

# Why Sampling?

- To learn about the population: population parameters
  - We don't get to measure/record/observe the *full population*, only a sample of it
- To allow greater attention to data exploration and data quality
  - For full data, it might be prohibitively expensive to:
    - Process missing values in data
    - Evaluate outliers
    - Meaningfully plot and visualize
- To provide scalability
  - Most algorithms scale non-linearly with data size
- To provide balanced group representations
  - Over-sampling of under-represented observations
  - Under-sampling of over-represented observation



# How to Characterize a Sample?

## Sample Statistic

- **Single sample:**
  - mean, median, standard deviation
  - proportions, ratio of proportions
- **Two samples:**
  - the difference in means
  - the difference in proportions
  - ratio of proportions
- **Proxy statistic:**
  - $t$-statistic
  - $F$-statistic
  - $\chi^2$-statistic
  - $Z$-statistics

# Sample Statistics vs. Population Parameters

## S.S. vs. P.P.

| Sample Statistics | S.S. | P.P. | Population Parameters |
|---|---|---|---|
| The mean of a quantitative variable within a sample | $\bar{x}$ | $\mu$ | The mean of a quantitative variable in an entire population |
| The standard deviation of a quantitative variable within a sample | $s$ | $\sigma$ | The standard deviation of a quantitative variable in a population |
| The variance of a quantitative variable within a sample | $s^2$ | $\sigma^2$ | The variance of a quantitative variable in a population |
| The proportion of an outcome occurring within a sample | $\hat{p}$ | $p$ | The proportion of an outcome occurring in a population |
| The proportion of something not occurring within a sample | $\hat{q}$ | $q$ | The proportion of something not occurring in a population |

Sample Statistics: Hats and Bars

# Population Parameters for Different Distributions

| Distribution | Degrees of freedom | Mean | Variance | Comments |
|---|---|---|---|---|
| Normal | | $\mu$ | $\sigma^2$ | |
| $t$ | $n$ | 0 | $n/(n-2)$ | |
| $F$ | $n_1$ and $n_2$ | $n_2/(n_2-2)$ | $a/b$ | $a = 2n_2^2(n_1+n_2-2)$<br>$b = n_1(n_2-2)^2\,(n_2-4)$ |
| $\chi^2$ | $r$ | $r$ | $2r$ | |

# Methods for Samples Drawn from Known Distributions

| Distribution | Random Variable Sample | Density | Probability |
|---|---|---|---|
| Normal | scipy.stats.norm.rvs() | scipy.stats.norm.pdf() | scipy.stats.norm.cdf() |
| $t$ | scipy.stats.t.rvs() | scipy.stats.t.pdf() | scipy.stats.t.cdf() |
| $F$ | scipy.stats.f.rvs() | scipy.stats.f.pdf() | scipy.stats.f.cdf() |
| $\chi^2$ | scipy.stats.chi2.rvs() | scipy.stats.chi2.pdf() | scipy.stats.chi2.cdf() |

*distribution_abbreviation.*{rvs/pdf/cdf} ()

- **rvs** = random variable (RV) sample generation
- **pdf** = probability density function of a given RV
- **cdf** = cumulative probability distribution function of a given RV

- scipy.stats.norm.**cdf(a)** $\equiv \boldsymbol{P}(\boldsymbol{X} \leq \boldsymbol{a})$: probability that $a$ or smaller number occurs in the normal distribution
- scipy.stats.norm.**cdf(b)** – scipy.stats.norm.**cdf(a)** $\equiv \boldsymbol{P}(\boldsymbol{a} \leq \boldsymbol{X} \leq \boldsymbol{b})$: probability that the variable falls between two values in the normal distribution

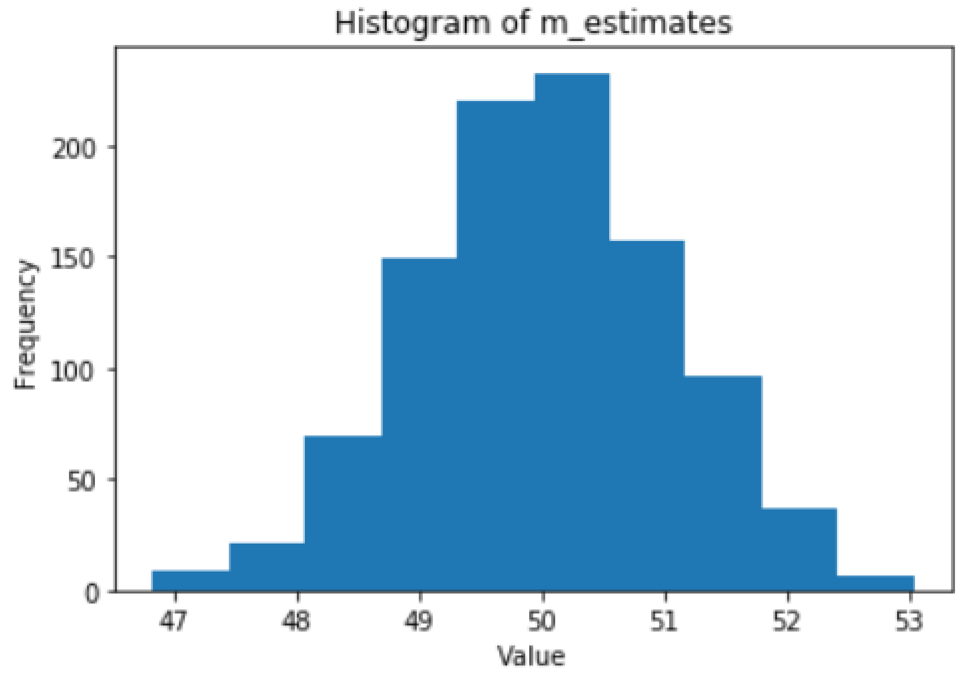
# Is the Sample Mean the same as the Population Mean?

## Population Parameters: mu and sd

```python
m_estimates = [stats.norm.rvs(loc=mean,
                              scale=sd,
                              size=sample_size).mean()
               for _ in range(n_samples)]

plt.hist(m_estimates)
plt.title("Histogram of m_estimates")
plt.xlabel("Value")
plt.ylabel("Frequency")
#plt.gcf()
plt.show()
```

sampling.ipynb

## Sample Statistic




**How sample statistic approximates population parameters for different sample sizes, n?**

```python
print ("Mean of sample means: ", np.array(m_estimates).mean())
print ("Standard Error: ", np.array(m_estimates).var())
```

```
Mean of sample means:  50.0111166372
Standard Error:  0.956456485027
```

# Ex: Sample from Unit Normal Distribution

$$N(\mu = 0, \sigma = 1)$$

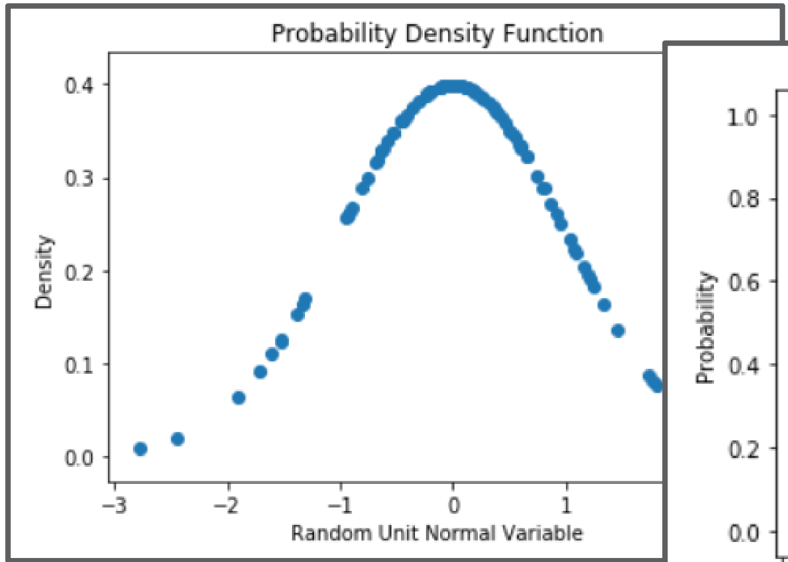
## scipy.stats.norm.pdf()

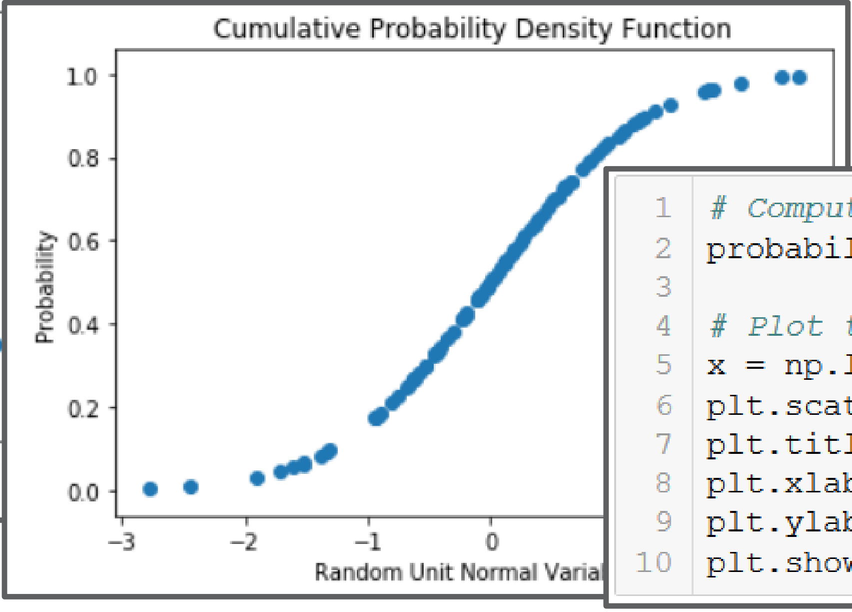
```python
# Calculate and plot their probability density functions
densityRandUnitNormal = stats.norm.pdf(randUnitNormal)

x = np.linspace(norm.ppf(0.01), stats.norm.ppf(0.99), 100)
plt.scatter(randUnitNormal, densityRandUnitNormal)
plt.title("Probability Density Function")
plt.xlabel("Random Unit Normal Variable")
plt.ylabel("Density")
plt.show()
```

## scipy.stats.norm.rvs()

```python
# Generate 1000 points drawn
# from the unit normal distribution: N( 1.0, 0.0)
mean = 0.0
sd = 1.0
randUnitNormal = scipy.stats.norm.rvs(loc=mean,
                                      scale=sd,
                                      size=100)

randUnitNormal[0:3]
```

```
array([-0.96907238, -0.24168709,  2.19288252])
```

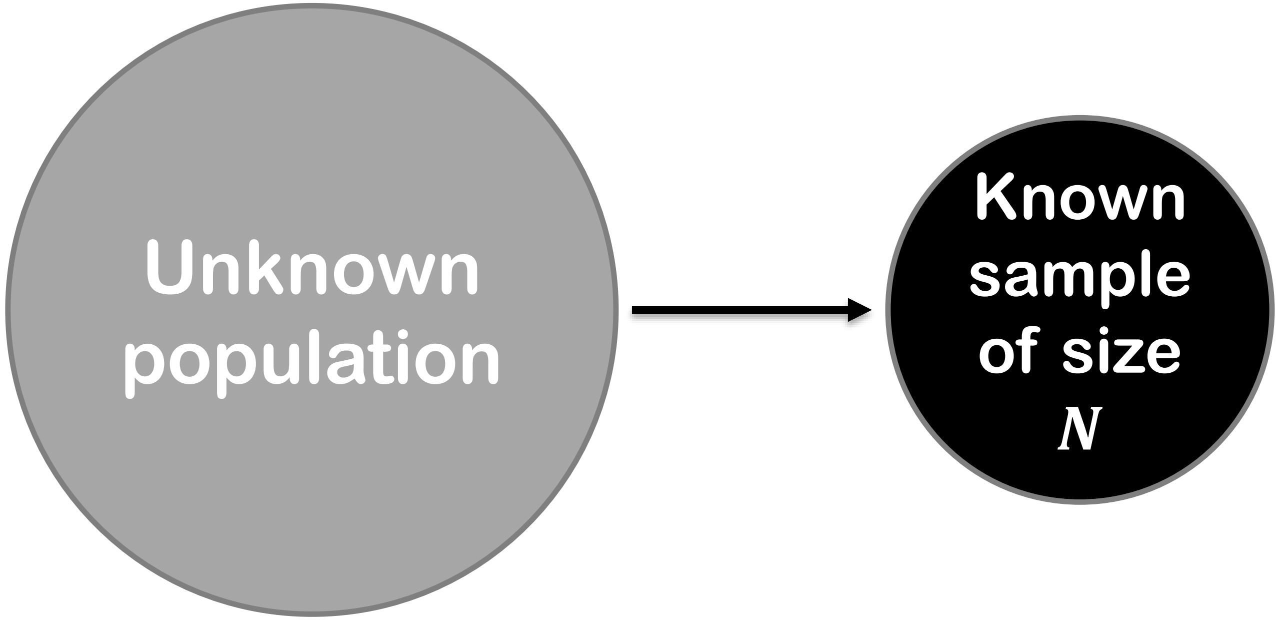



## scipy.stats.norm.cdf()

```python
# Compute and plot cumulative probability distribution
probabilityRandUnitNormal = stats.norm.cdf(randUnitNormal)

# Plot the distribution
x = np.linspace(norm.ppf(0.01), stats.norm.ppf(0.99), 1000)
plt.scatter(randUnitNormal, probabilityRandUnitNormal)
plt.title("Cumulative Probability Density Function")
plt.xlabel("Random Unit Normal Variable")
plt.ylabel("Probability")
plt.show()
```

# Sample Drawn from an Unknown Population




- How do samples drawn from an unknown population behave?
  - How different are they from one another?

# Statistic & its Proxy: Hypothesis Testing

| Aim | Model Statistic | Sample Statistic | Proxy Statistic | Formula for Proxy |
|---|---|---|---|---|
| Estimate the mean $\mu$ of a normal distribution with known variance $\sigma^2$ | $\mu$ | $m$ | $Z$-statistic | $Z \sim \frac{m-\mu}{\sigma/\sqrt{n}}$ |
| Estimate the variance $\sigma^2$ of a normal distribution with known mean $\mu$ | $\sigma^2$ | $S^2$ | $\chi^2$-statistic | $\chi^2_{n-1} \sim (n-1)\frac{S^2}{\sigma^2}$ |
| Estimate the mean $\mu$ of a normal distribution with un-known variance $\sigma^2$ | $\mu$ | $m$ | $t$-statistic | $T_{n-1} \sim \frac{m-\mu}{S/\sqrt{n}}$ |

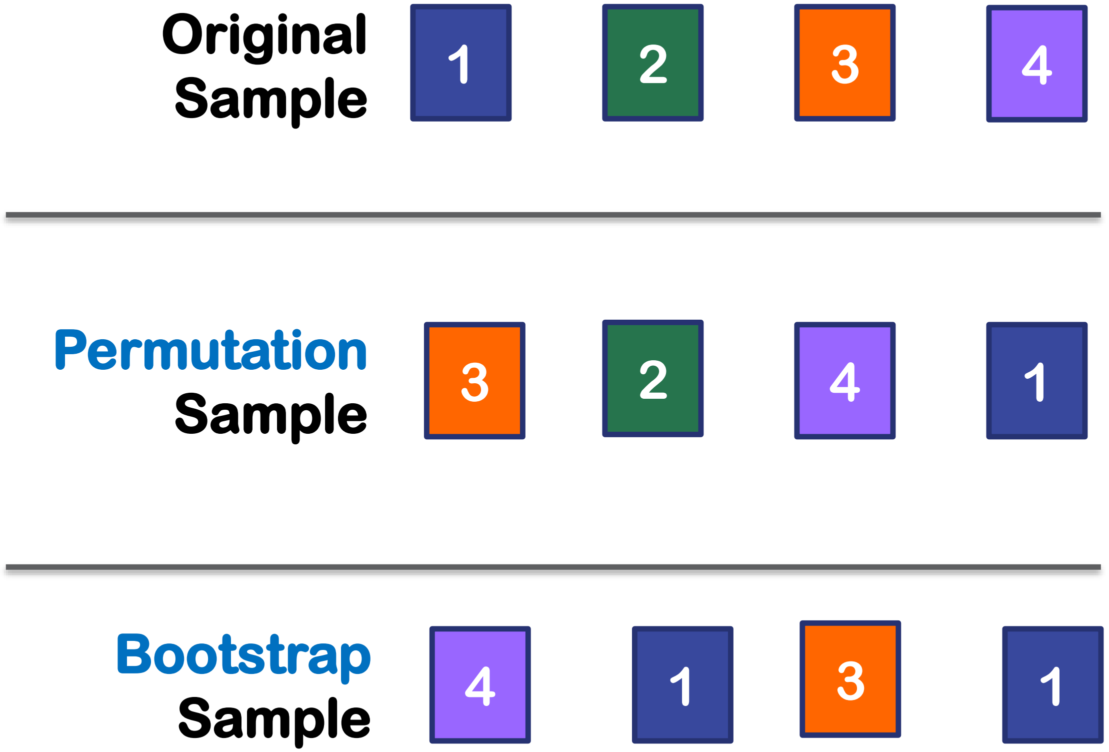
| Ex. | Proxy Statistic | Distribution | Degrees of Freedom (df) |
|---|---|---|---|
| 1 | $Z$-statistic | $N(0,1)$ | |
| 2 | $\chi^2$-statistic | $\chi^2(n-1)$ | $n-1$ |
| 3 | $t$-statistic | $T_{n-1}$ | $n-1$ |

# Sampling Schemes

## RESAMPLING, BOOTSTRAP & PERMUTATION SAMPLING

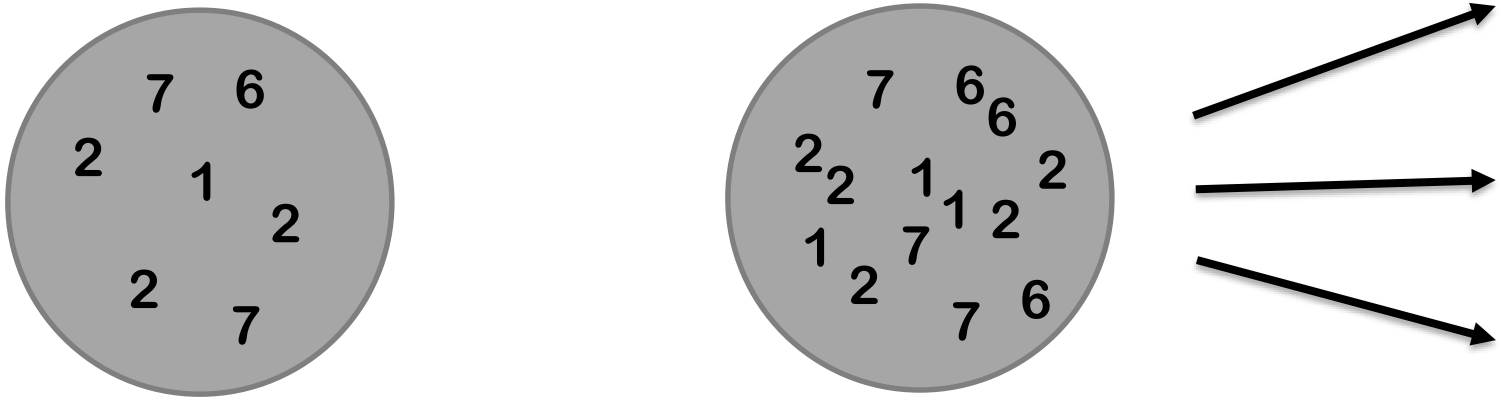
# Resampling: Bootstrap and Permutation

- **Bootstrap Sampling**:
  - Sampling with replacement
  - Hypothesis Testing
  - Confidence Interval Estimation
  - Python package: bootstrap-tools
  - http://gcalmettes.github.io/bootstrap-tools/
- **Permutation Sampling**:
  - Sampling without replacement: shuffling
    - numpy.random.permutation(x)
    - numpy.random.shuffle(x)
  - Permutation Tests: Independence Problems
    - Python package: pip install permute
    - Are responses independent of group labels?
    - Are two/k samples independent?
    - Are two categorical variables independent?
  - Permutation Tests: ANOVA & Regression Designs
    - Define later when we study regression



# Basic Bootstrap: Theory

Hypothetical Population

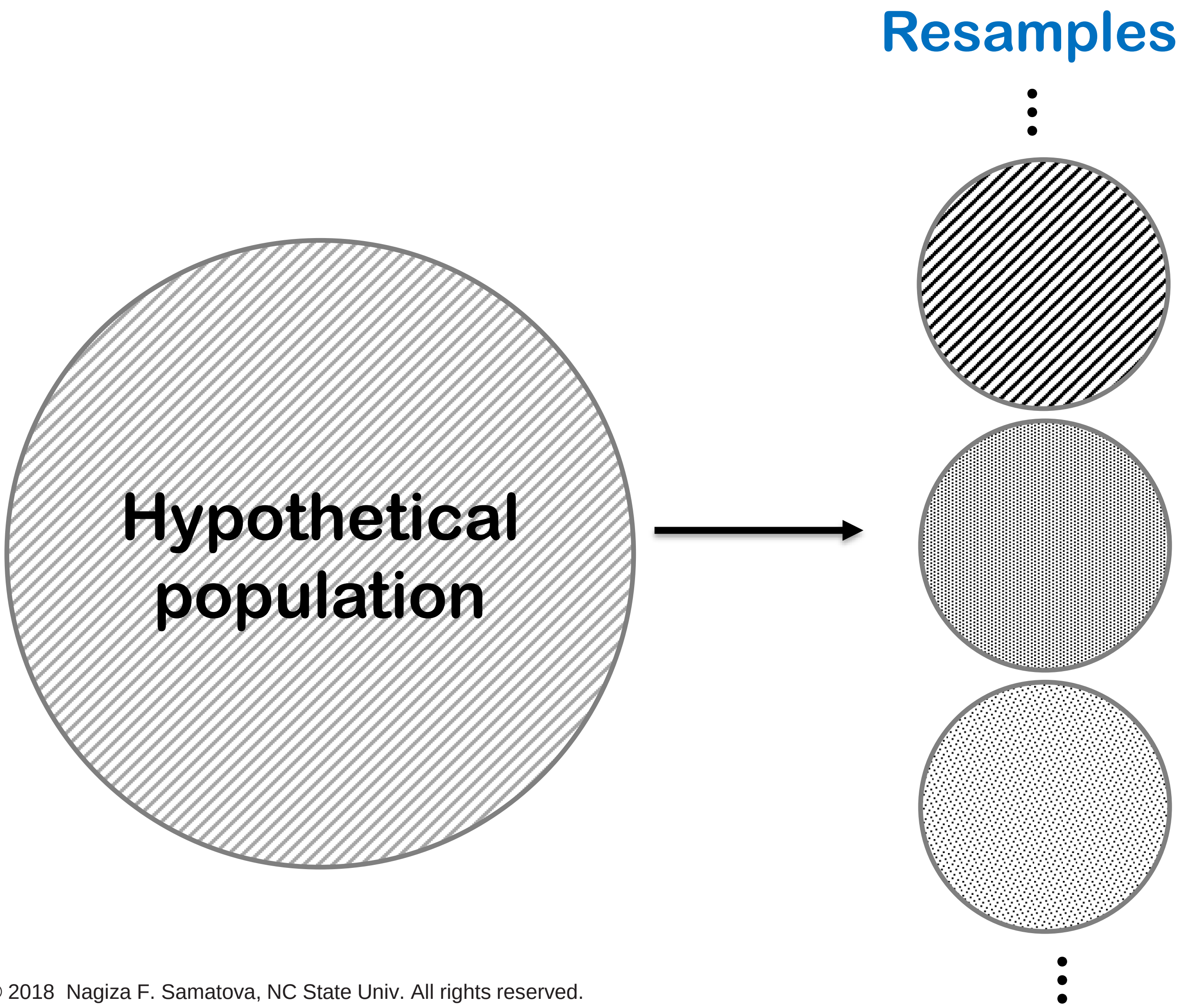



Original Sample

Sample replicated a huge number of times

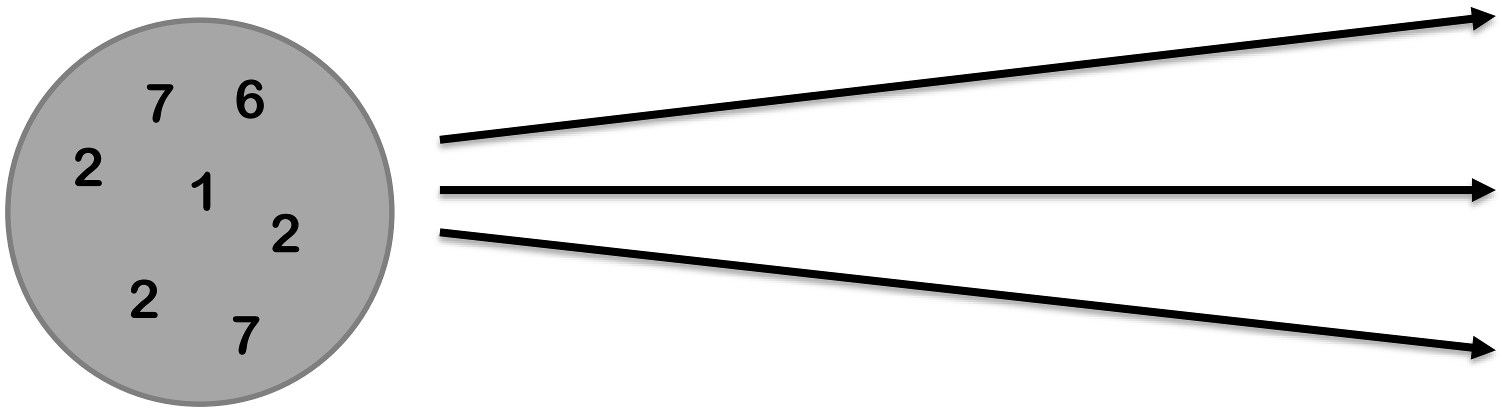
Draw lots of resamples

# Simulation: Bootstrap Sampling Procedure: In Theory




1. From the observed known sample, calculate a statistics to measure some attribute of the population (e.g., positive response rate, mean)
2. Create a hypothetical population using information from the sample
3. Draw a resample from the hypothetical population
4. Record the statistic of interest for the resample
5. Repeat steps 3 and 4 many times
6. Observe the sampling distribution of the statistic of interest to estimate an error or difference from the benchmark value of interest

# Basic Bootstrap: Practice

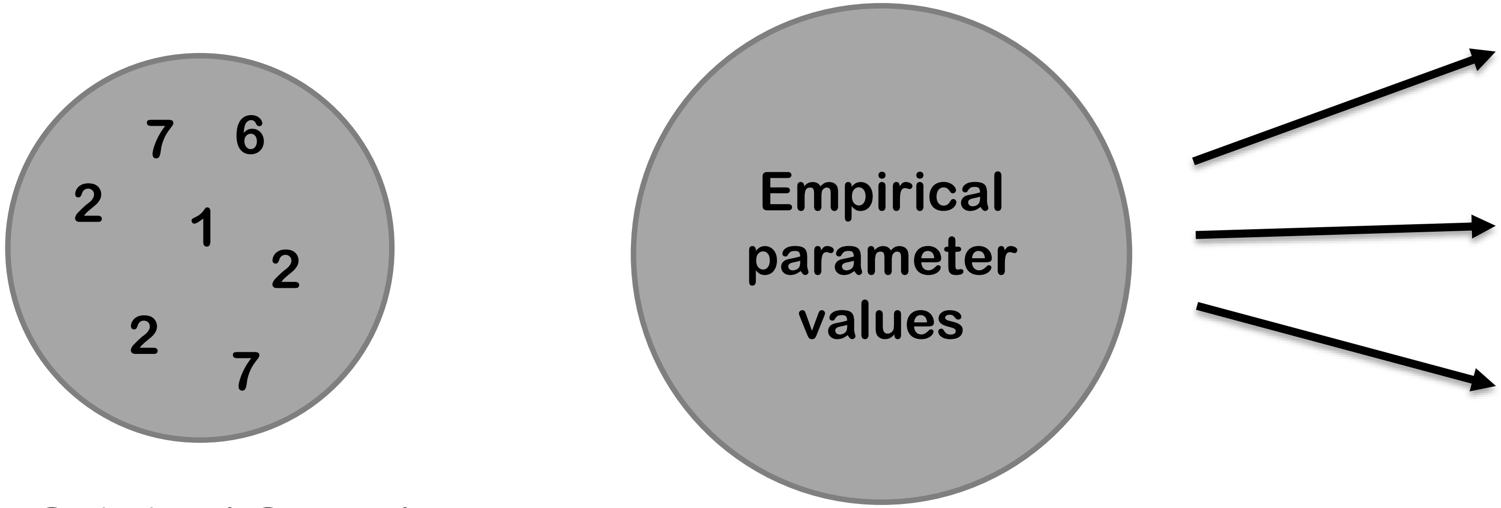



Original Sample

Draw lots of resamples,
with replacement

# **Parametric Bootstrap**

Known Distribution: Population




Original Sample

Random number generator

Draw lots of resamples,

- Normal distribution parameters
  - $\bar{x}$ : mean from the sample
  - $s$ : standard deviation from the sample

# Bootstrapping with the bootstrap-tools.bootci() in Python

**ci = bootstrap-tools.bootci (data = , stat = , ...)**

1. Write a function (e.g., statistic_funct()) that returns the statistic or statistics of interest
2. Pass this function to the .bootci() as statistic = statistic_function
3. Pass the number nboot of bootstrap replicates
4. Use .bootci() method to obtain confidence intervals for the statistic(s) generated in Step 2

```
Signature: bootci(data, stat=<function median at 0x0000000006EDB158>, nboot=1000, replace
ment=True, alpha=0.05, method='pi', keepboot=False)
Docstring:
Compute the (1-alpha) confidence interval of a statistic (i.e.: mean, median, etc)
of the data using bootstrap resampling.

Arguments:
    stat:        statistics we want the con
    nboot:       number of bootstrap sample
    replacement: resampling done with (True
    alpha:       level of confidence interv
    method:      type of bootstrap we want
    keepboot:    if True, return the nboot
                 the confidence intervals a
```

```python
loans_income = pd.read_csv("../data_raw/sampling_loans_income.csv")
ci = bootci(data = loans_income,
            stat = np.median,
            alpha = 0.05)

print ("Estimate of the Median Income: ", loans_income.median())
print ("The 95% confidence interval for the estimated median: ", ci)
```

```
Estimate of the Median Income:  x    62000.0
dtype: float64
The 95% confidence interval for the estimated median:  (61000.0, 62000.0)
```

# Sampling Strategies

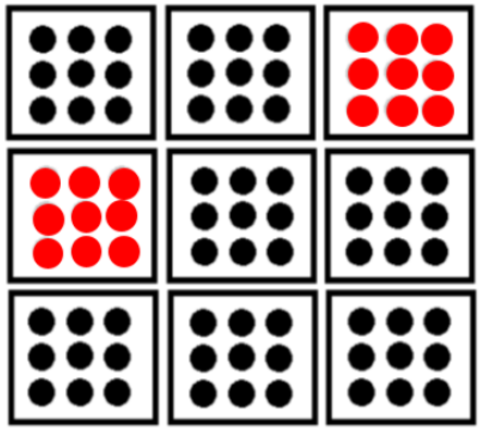
## TYPES OF SAMPLING

# Sampling Strategies

- Simple Random Sample
- Stratified Random Sample
- Cluster Sample
- Systematic Sample

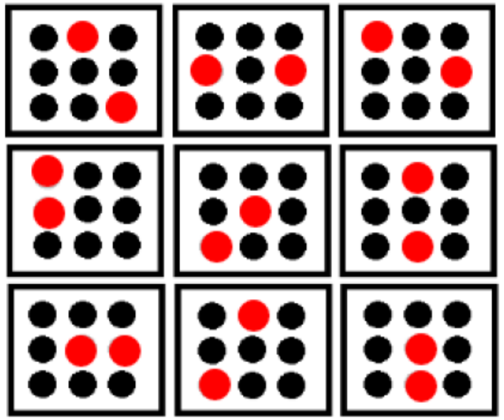
# Sampling Strategies: Visual Illustration

### Cluster



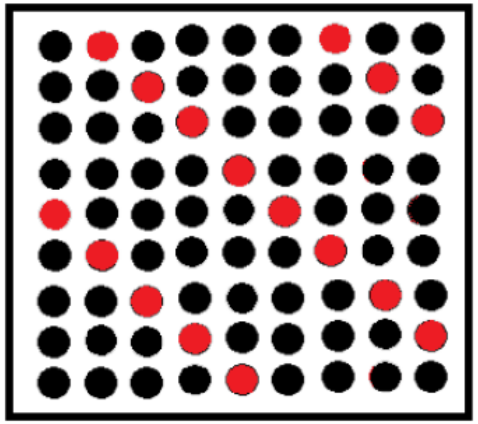
Randomly select 2 clusters and sample every individual in those

### Stratified



Randomly select 2 individuals from each strata

### Systematic



Randomly select $2^{nd}$ individual, then select every $5^{th}$ individual after that

# Sampling Strategies

| Term | Definition | Pros and Cons |
|---|---|---|
| Convenience Sampling | There is no effort to define a population or sampling frame: by inviting any one who saw the invite | (+) Easy and cheap<br>(-) Non-representative sample, not well-designed |
| Cluster Sampling | Clusters of subsects or records selected, and the subjects or records within those clusters are surveyed and measured. Ensure that characteristics that define clusters do not introduce bias into the results | (+) Practical and efficient |
| Multi-stage Sampling | Randomly select groups and then apply systematic sampling within each group | (+) Minimize cost, sampling error, and bias |
| Self-Selection | The respondents themselves determine whether they participate in the survey | (-) Biased results |
| SRS: Simple Random Sample | Better known as a randomly drawn sample rather than random sample: each object in the population has an equal chance of being selected | (-) Does not guarantee a fully representative sample<br>(-) Inefficient in practice |
| Stratified Sampling | The population is split into categories, or strata, and separate samples are drawn from each stratum. | |
| Systematic Sampling | Selection of every $n^{th}$ record | |

# SRS: Simple Random Sample (pd.DataFrame.sample())

sampling.ipynb

```
file = "../data_raw/sampling_customer_satisfaction.csv"
cust_sat = pd.read_csv(file)
cust_sat.head(2)
```

| | ID | size | sizecount | sc |
|---|---|---|---|---|
| 0 | 1027 | Medium | 500 | 60.4579 |
| 1 | 257 | Large | 500 | 60.3947 |

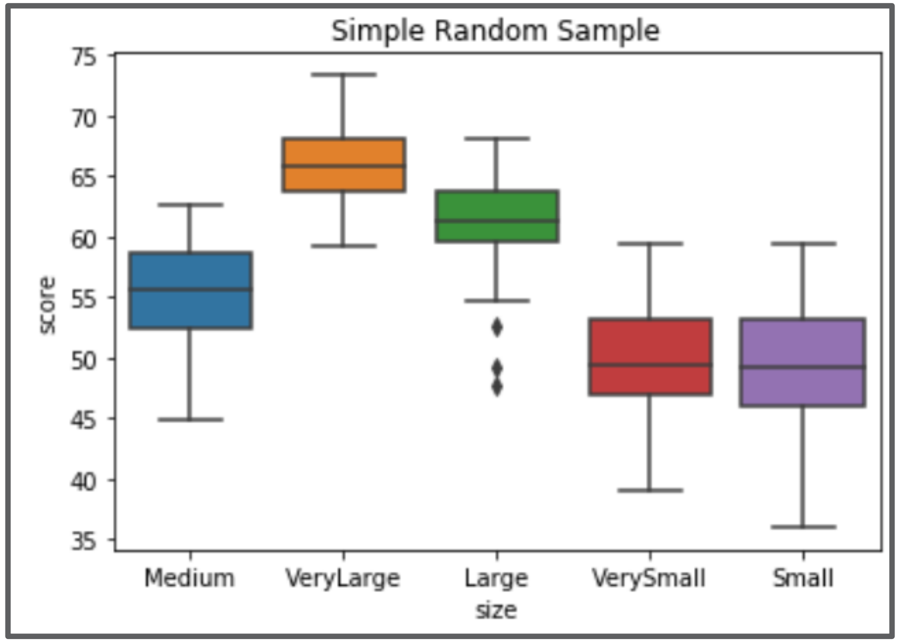
```
cust_sat_srs = cust_sat.sample(
    n=200, replace=False)
cust_sat_srs.shape
```

```
(200, 4)
```

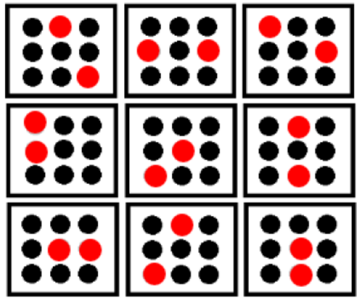
- **Assumptions**
  - population: homogeneous
- **Pros**
  - Simple in theory
  - Unbiased
  - Makes statistical inference possible
- **Cons**
  - Complex or inefficient in practice
  - Does not guarantee a completely random sample
- **Python Examples:**
  - DataFrame.sample()

```
sns.boxplot( x="size",
             y="score",
             data = cust_sat_srs )
plt.show()
```

**Bootstrap Sample without Replacement**

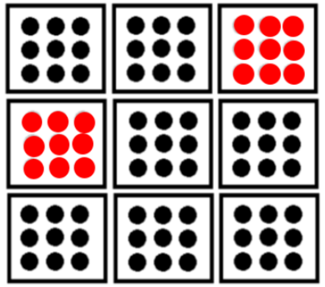


# Stratified Sampling



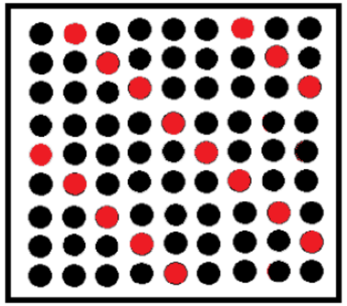
- **Assumptions**
  - population is divided into subgroups called strata
  - with important differences across strata
- **Pros**
  - usually increases precision
  - allows separate estimates per stratum
  - convenient/easier/cheaper
- **Cons**
  - requires knowledge of auxiliary variable
  - complicates analysis
- **Example**
  - Customer satisfaction:
    - Want to get input from different-sized customer orgs, different sectors, different regions

# Cluster Sampling



- **Assumptions**
  - observational units are not directly accessible:
    - **SRS of customer organizations**
    - **then SRS of employees within selected organizations**
  - clusters are representative of populations
- **Pros**
  - cheaper, easier, more convenient than SRS
  - only need a list of clusters (not all observations)
- **Cons**
  - strong dependence within clusters may lead to inefficiency
  - more complex analysis than SRS

# Systematic Sampling



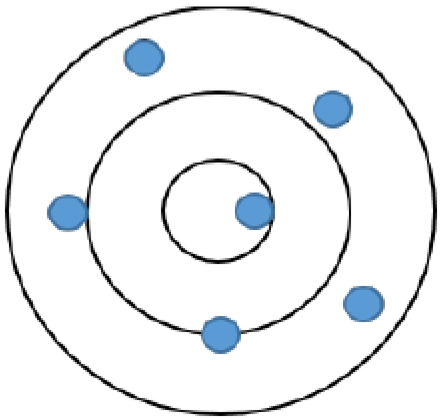
- **Assumptions**
  - population is homogenous or
  - strata/clusters are systematically arranged
- **Pros**
  - easy to implement
  - useful for data over time
  - convenient/cheap
- **Cons**
  - can be biased if not carefully selected
    - **seasonality, periodicity**
  - accuracy depends on the order of sampling units; never an SRS
- **Example**
  - Quality Control
    - Sample every $100^{th}$ item one item per hour from a continuous moving production line

# Sampling

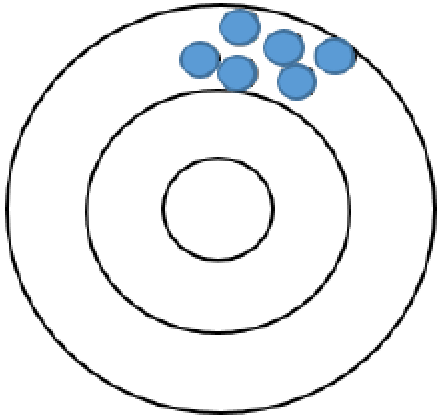
## SAMPLE DESIGN

# Representative Sample that leads to Accuracy & Precision

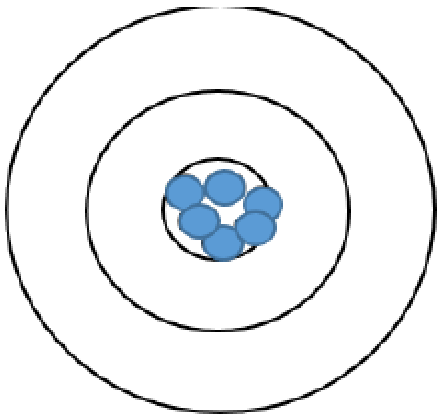
**A small representative sample is more accurate and precise than a large sample that is not representative**



**Accurate**
**Not Precise**



**Not Accurate**
**Precise**



**Accurate**
**Precise**

# Sample Characteristics: Accuracy and Precision

- **Accuracy**
  - Mean
  - Median
  - Mode
- **Precision**
  - Variance
  - Interquartile range
  - Mean Absolute Deviation
- **Bounds on the Error of Population Parameter Estimation**
  - E.g., the probability that sample mean is different from the population mean within a given error is 0.95

# Sample Design Goal & Criteria for Good Design

- Goal:
  - Maximize information while minimizing cost

- Criteria
  - Accuracy: how far is sample statistic from the corresponding population parameter (P.P.)
  - Precision: how small is standard error for a sample statistic
  - Error bounds: how small is the error on the P.P. estimation

# Sample Design Procedure

- Design Process
  - Step-1: Decide on sampling strategy
  - Step-2: Select sample size
    - **Power Analysis slides on the Sample Size selection**

# Step-1: Decide on Sampling Strategy
## General Guidelines

- **Use stratified sampling**
  - To insure representation from particular groups
- **Use cluster sampling**
  - If individuals are spread out geographically or
  - If information/time/money is limited
- **Use systematic sampling**
  - If need to measure in real time
- **Context and pragmatism are key**
  - "Perfect" sampling plan no good if it cannot be implemented

# Step-1: Decide on Sampling Strategy
## Other Considerations

- How are individuals organized in the population?
  - What information is available?
  - Can I get a sampling frame for all individuals, or do I only have a list of clusters?
- How much time/money/resources can be devoted to collecting data?
- What do I want to learn about?
- Don't sample based on a response variable:
  - Want to measure customer satisfaction, but only sample from customers with historically high ratings is available

# Sampling Bias

## SELECTION BIAS AND RESPONSE NATURE

*Biased samples are more likely to produce some outcomes than others… sample statistics may be consistently too high or too low*

# Bias due to Selection or the Nature of the Response

| Term | Definition | Examples/Comments |
|---|---|---|
| Bias | A statistical procedure or measure is biased if applied to a sample from a population produces (under-)over-estimates of population characteristic | |
| Nonresponse Bias | A problem that occurs when non-responders do not show up in surveys | |
| Response Bias | Responses given differ from the truth | |
| Self-Selection | The respondents themselves determine whether they participate in the survey | (-) Biased results |
| Convenience Sampling | There is no effort to define a population or sampling frame: by inviting any one who saw the invite | (+) Easy and cheap<br>(-) Non-representative sample, not well-designed |
| Selection Bias | Only a particular subset of people are selected or volunteer to be in the sample | |
| Volunteer response sample | Self-selected sample of people who responded to a general appeal | |

# Bias: Sample Selection

- **Selection bias**
  - Only a particular subset of people are selected or volunteer to be in the sample
- **Convenience samples**
  - Samples that are easy to take, based on a readily assembled group
    - **E.g., only selecting customers from a particular organization**
- **Volunteer response sample**:
  - Self-selected sample of people who responded to a general appeal
    - **Those who volunteer may be different from general population**
    - **Ex: Table cards in restaurants, online votes**
    - **Ex: Sending a general email blast to all customers**

# Other Sources of Bias

- **Non-response bias**
  - Some part of the population may not respond or refuses to participate
  - Connection to missing data:
    - **If responses are MAR (Missing at random), could impute**
    - **If MNAR (Missing not at random), a small response rate could indicate a problem**
- **Response bias**
  - Responses given differ from the truth
  - Results from questions or people involved; could be intentional or unintentional
    - **Ex: Customer may not want to mention in person that they are not satisfied**

# Other Things to Keep in Mind

- **It is important to pay attention to the sampling method used when considering the results of a survey**
- **If the sample is not random, proceed with extreme caution!**
  - You may not be able to make any conclusions about the full population
  - Instead, you have to think about what restricted/other population the sample is representative of