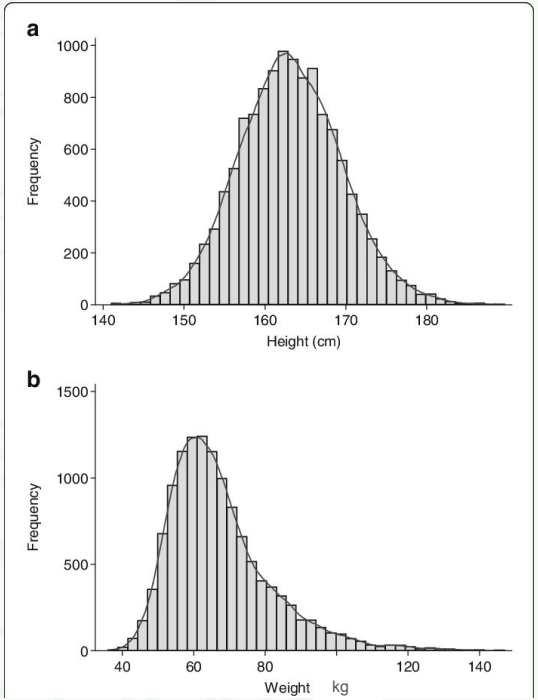
Sampling Data, Imbalance and Missingness

Prof. Murillo
Computational Mathematics, Science and Engineering
Michigan State University

Sampling Distributions Using Data

Goal: model a population of people with realistic heights and weights.

We need to model 2000 people.



Solution 1: sample a Gaussian/normal distribution to assign a height and weight to each person.

Problem 1: These distributions are ***not*** Gaussian/normal.

Problem 2: There are ***correlations*** between height and weight.

Solution 2: ***Data Science!***

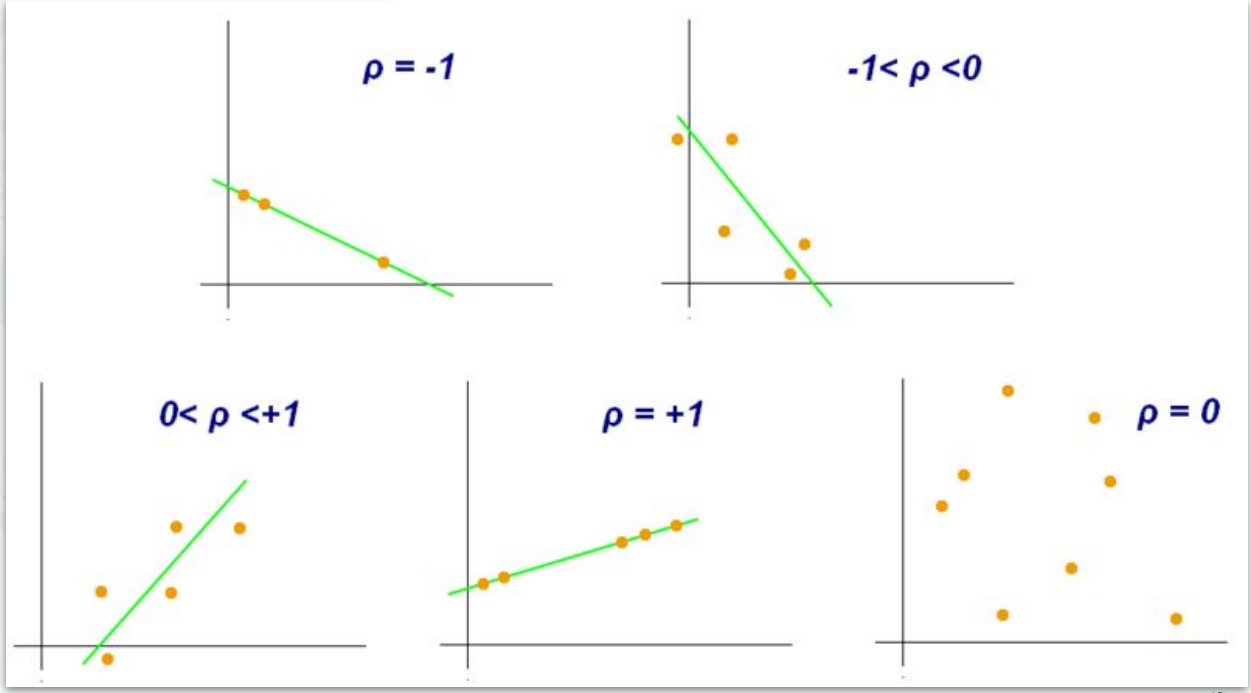
Correlations and Covariance

$$\operatorname{cov}(X,Y) = \operatorname{E}\left[(X - \operatorname{E}[X])(Y - \operatorname{E}[Y])\right]$$

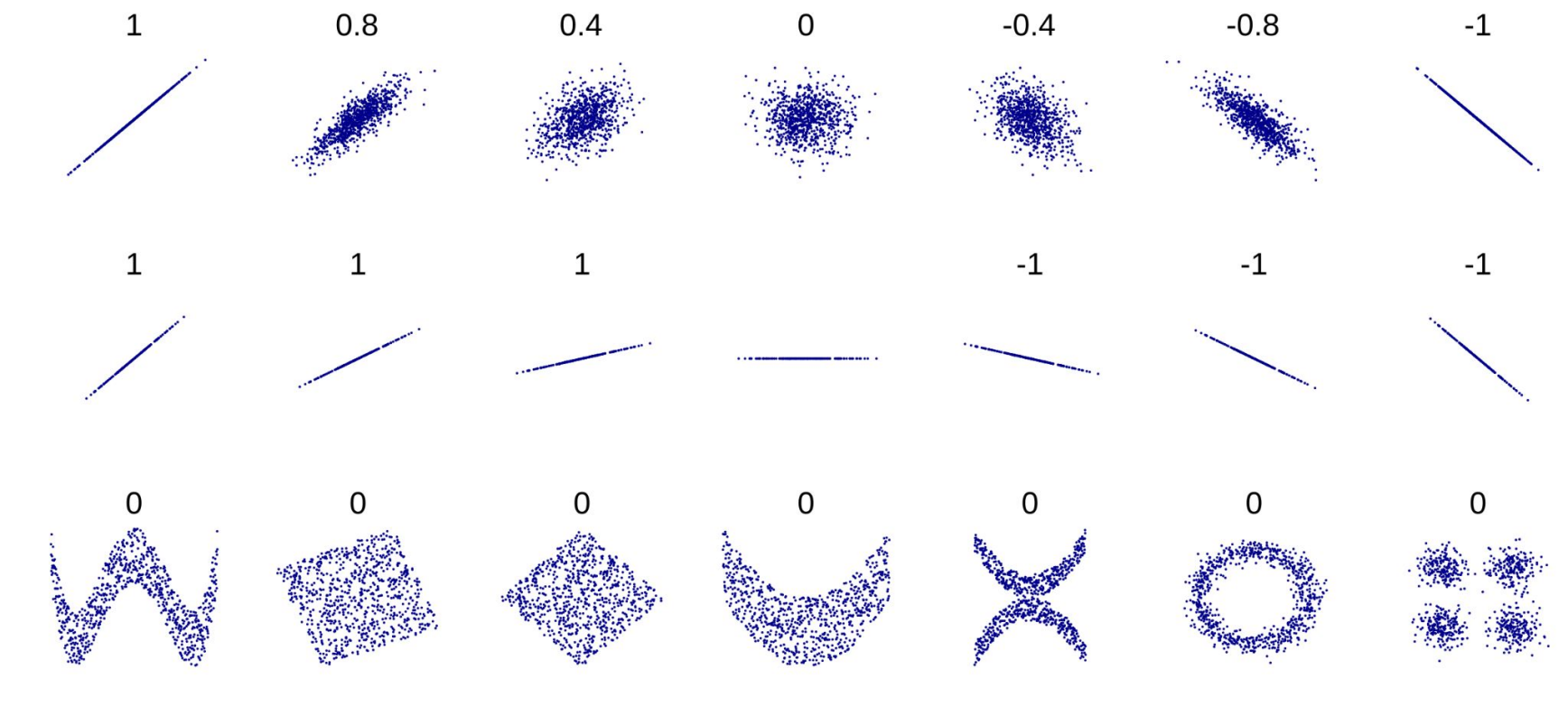
covariance

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

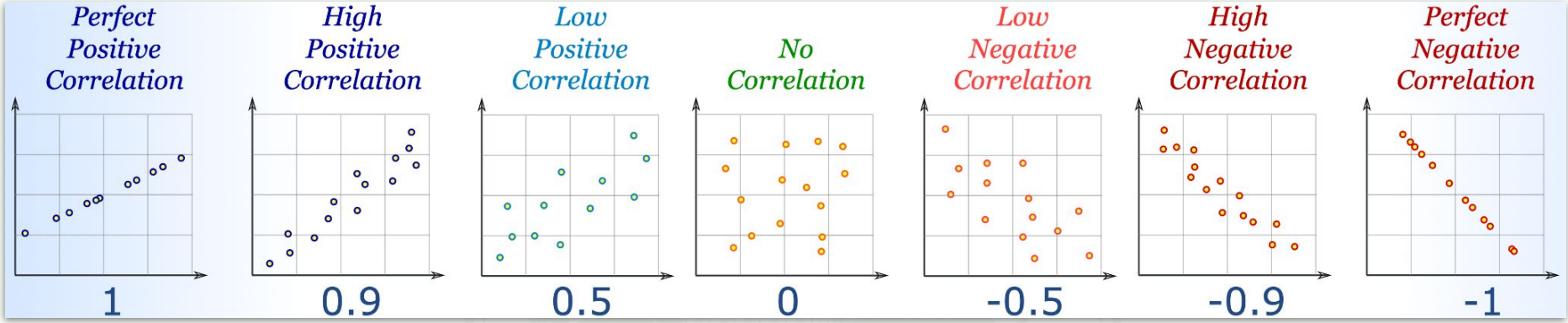
Pearson correlation



Correlations and Covariance



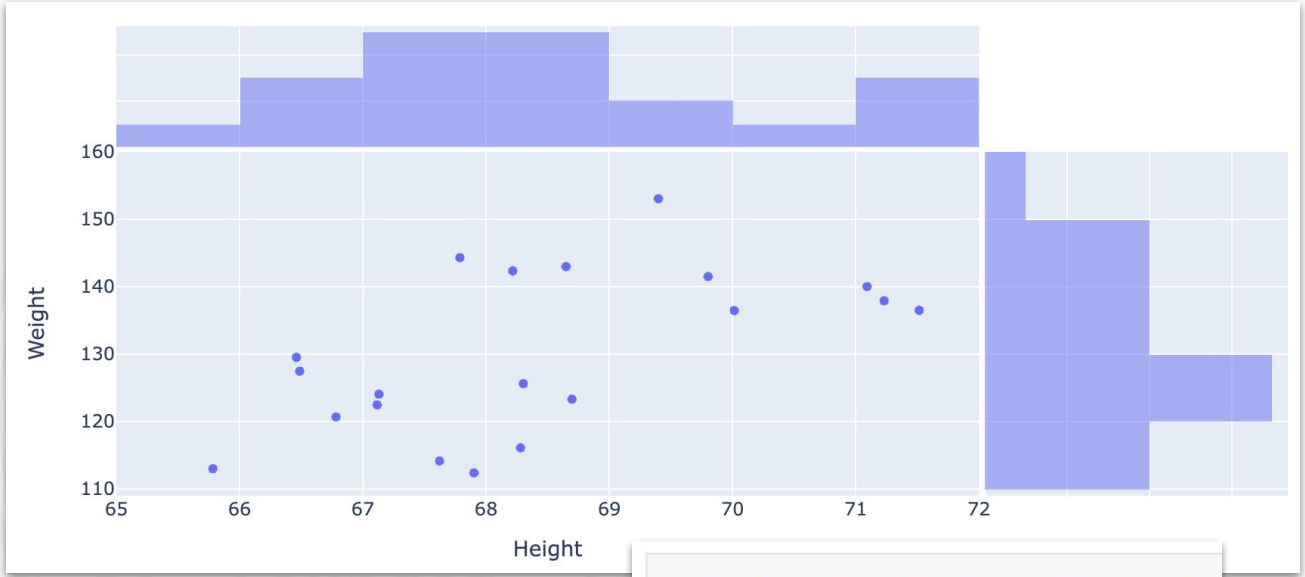
Words to Describe Correlation



$$\rho_{X,Y} = \frac{\mathbb{E}[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

Data on Heights and Weights

# Index Serial Number for the Dataset.	# Height(Inches) Height in Inches	# Weight(Pounds) Weight in Pounds
1 … 25.0k	60.3 … 75.2	78 … 171
1	65.78331	112.9925
2	71.51521	136.4873
3	69.39874	153.0269
4	68.2166	142.3354
5	67.78781	144.2971
6	68.69784	123.3024
7	69.80204	141.4947
8	70.01472	136.4623
9	67.90265	112.3723
10	66.78236	120.6672
11	66.48769	127.4516
12	67.62333	114.143
13	68.30248	125.6107
14	67.11656	122.4618
15	68.27967	116.0866
16	71.0916	139.9975
17	66.461	129.5023
18	68.64927	142.9733
19	71.23033	137.9025
20	67.13118	124.0449



This is a good use and interpretation of a *scatterplot*.

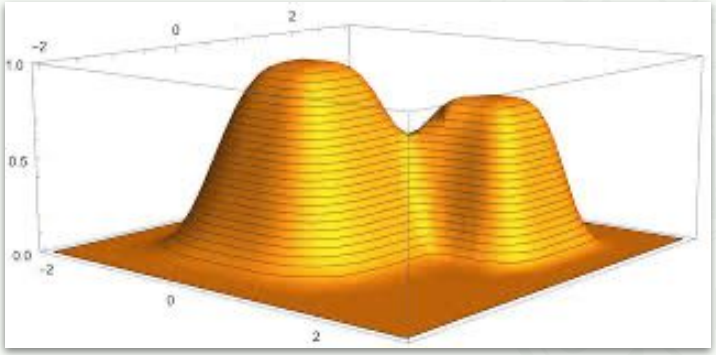
```
1  (df[["Height","Weight"]]).corr()
```

✓ 0.0s

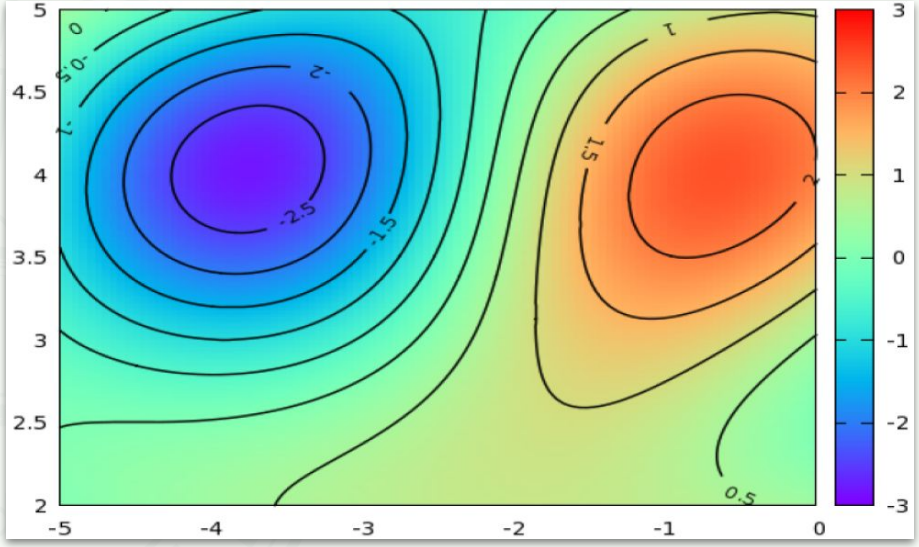
	Height	Weight
Height	1.000000	0.570459
Weight	0.570459	1.000000

Surface, Image and Contour Plots

To describe the probability of having a certain height and weight we need to add another dimension: `P(H,W)`.



This is the type of function we are after. We can plot this as a *surface* plot with *contours*.



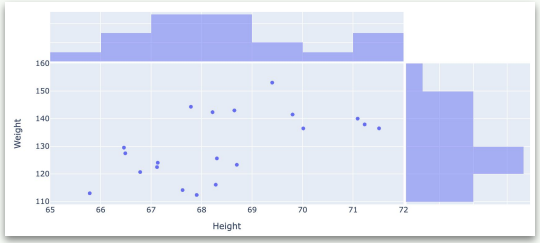
We can also show this in 2D by making the function value the color, and still include the contours.

Sampling Heights and Weights, Given Data

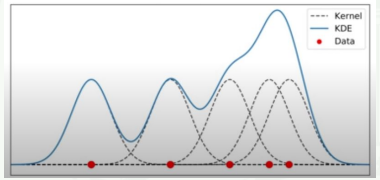
We (generally) need a smooth function for all values of the input parameters.

How can we do this?!

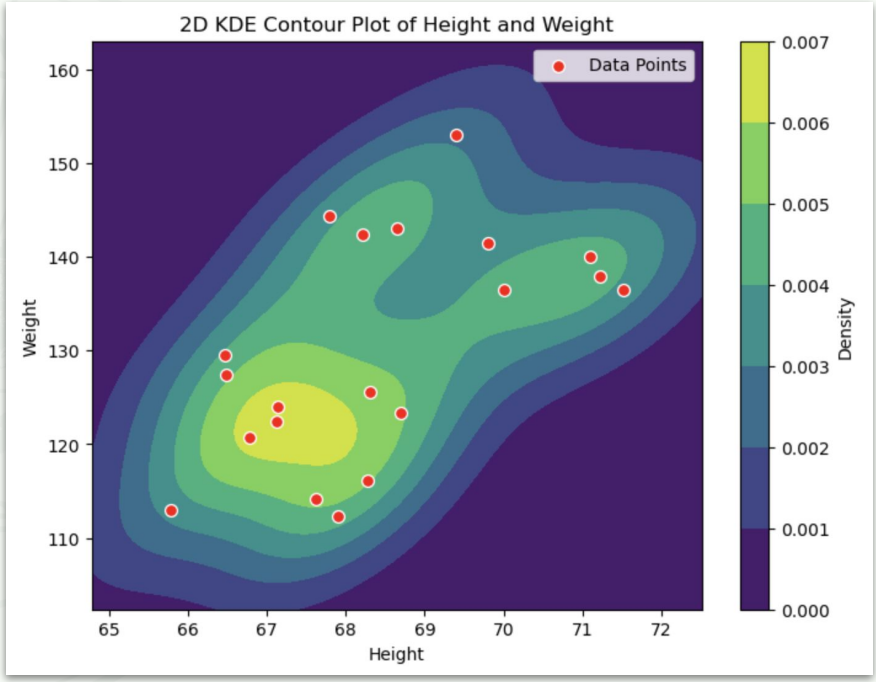
KDE.



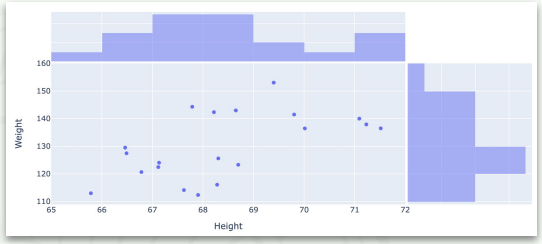
$$\hat{f}(x) = \frac{1}{N}\sum_{i=1}^{N} K(x - x_i).$$



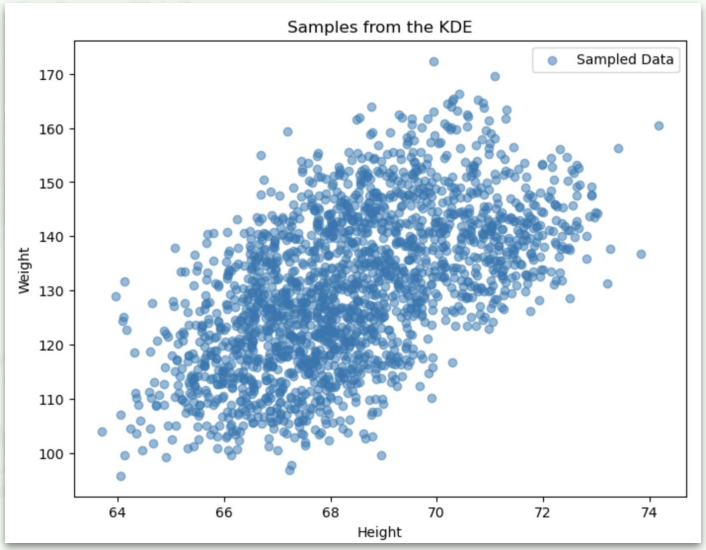
Put a kernel (usually a Gaussian) on each of these points and add them up.



Sample the KDE



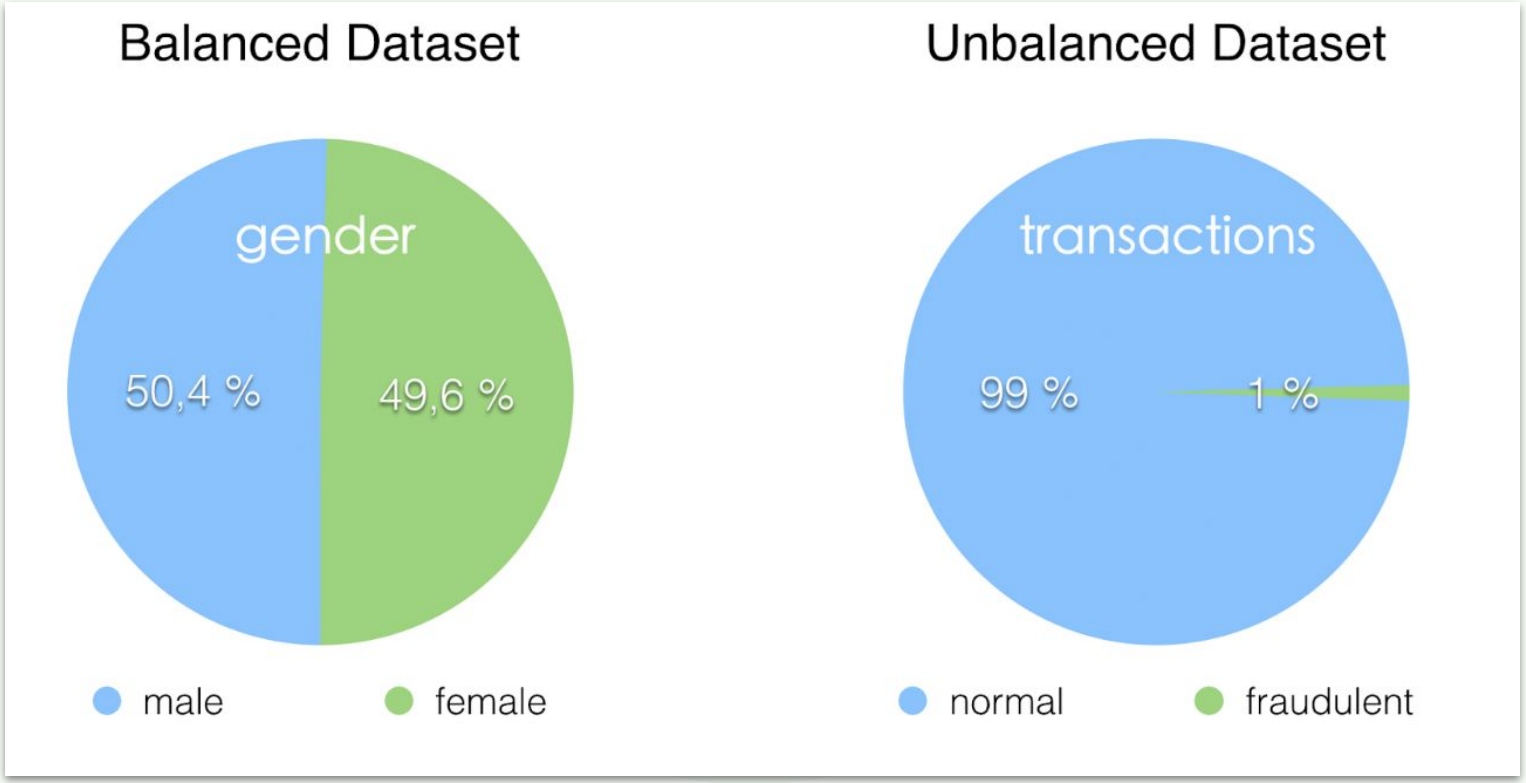
original data



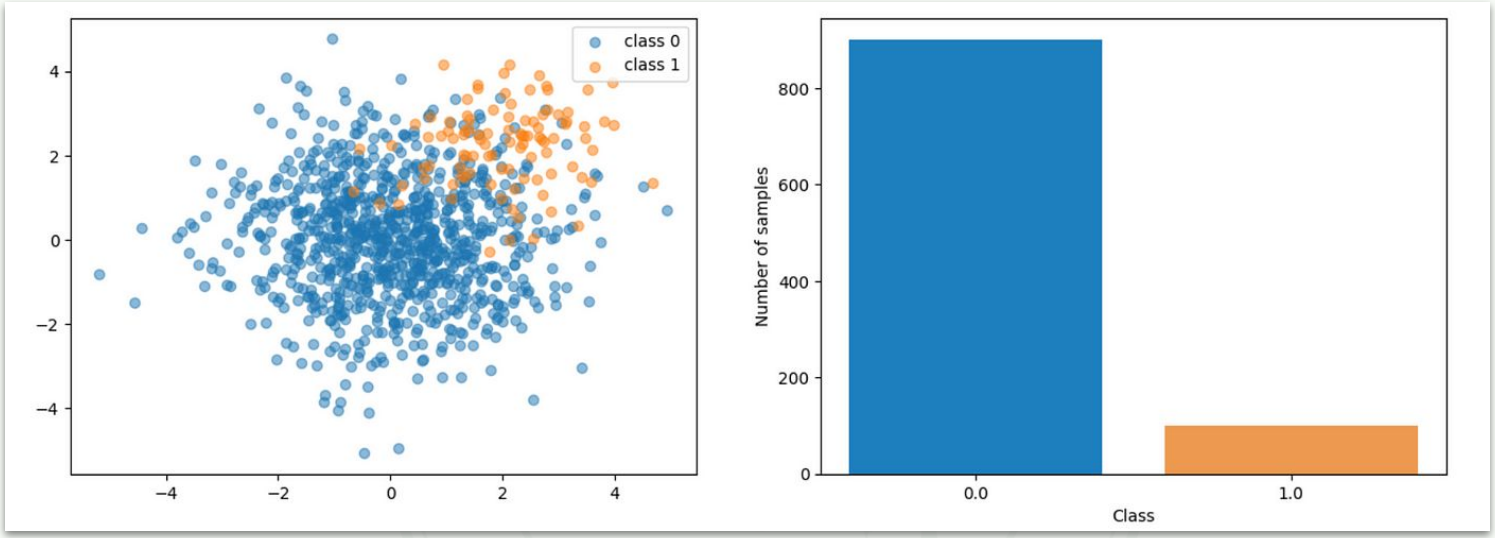
2000 samples

Unbalanced Data

Imbalanced Data: classes not represented equally



Problems with and Causes of Imbalanced Data

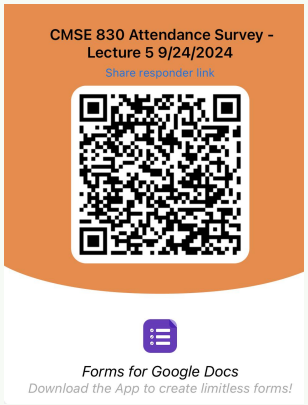


- Biased model performance
- Overfit to majority class
- Poor generalization
- Misleading evaluation metrics

- Natural imbalance in the problem domain
- Data collection bias
- Rare events or anomalies
- Limited access to minority class samples

Ethical Issues

- Fairness and discrimination concerns
- Misrepresentation of underrepresented groups
- Potential for reinforcing existing biases
- Impact on decision-making in critical domains (e.g., healthcare, criminal justice)



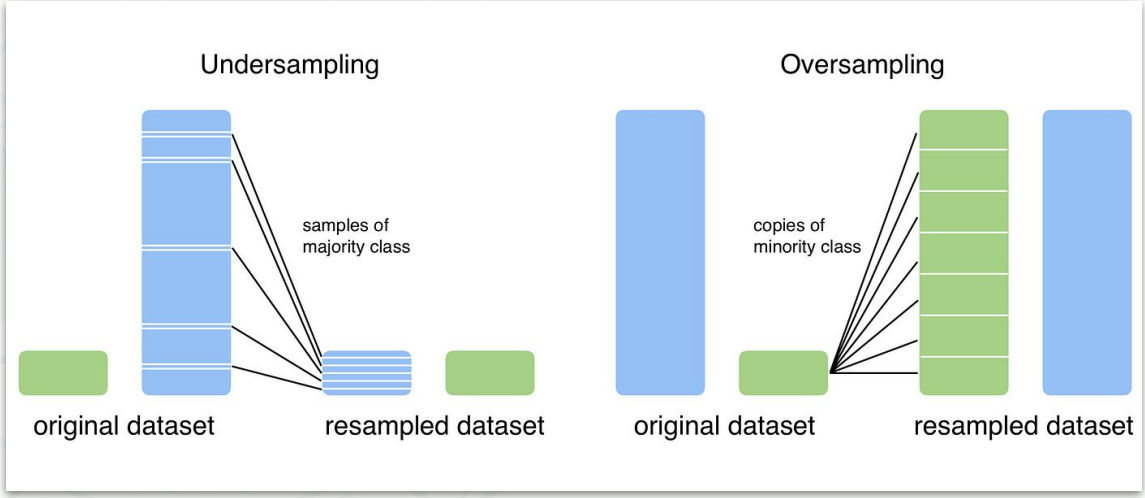
Addressing Imbalanced Data

data-level methods

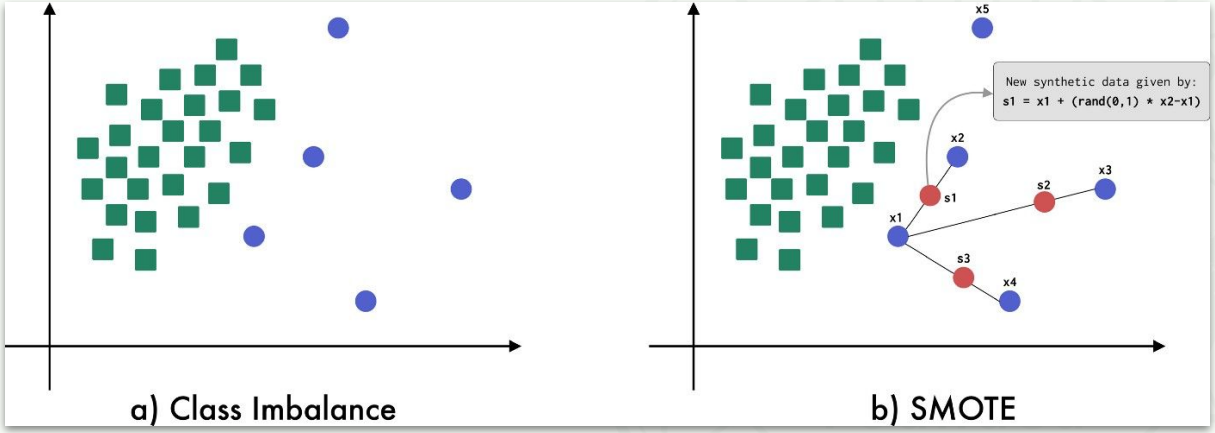
- oversampling minority class
- undersampling majority class
- synthetic data generation

algorithm-level methods

- adjust class weights
- ensemble methods
- one-class classification

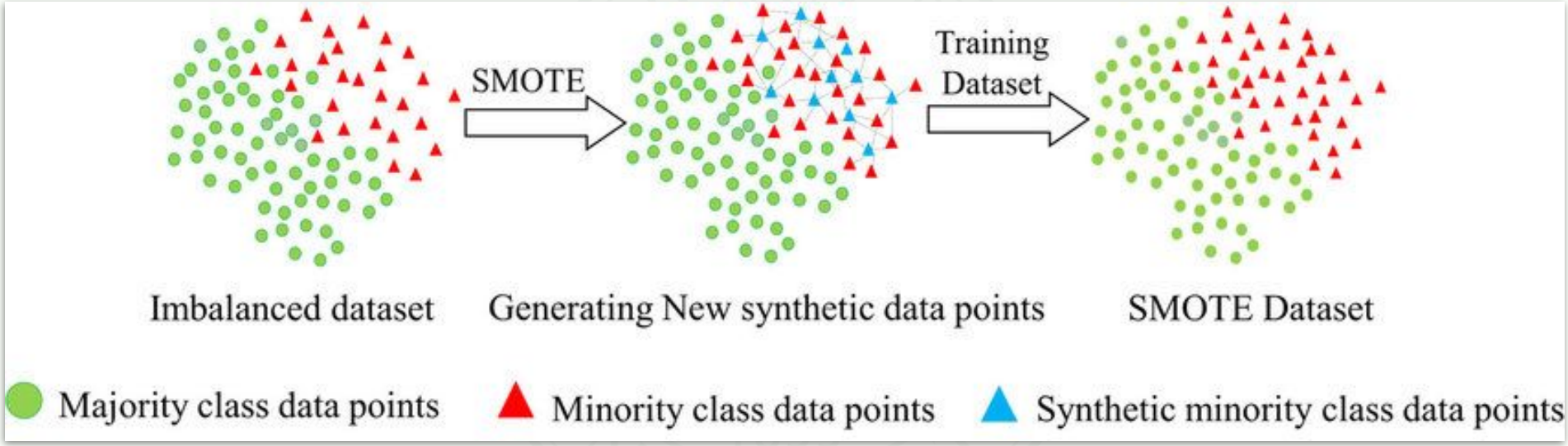


SMOTE: **S**ynthetic **M**inority **O**ver-Sampling **Te**chnique

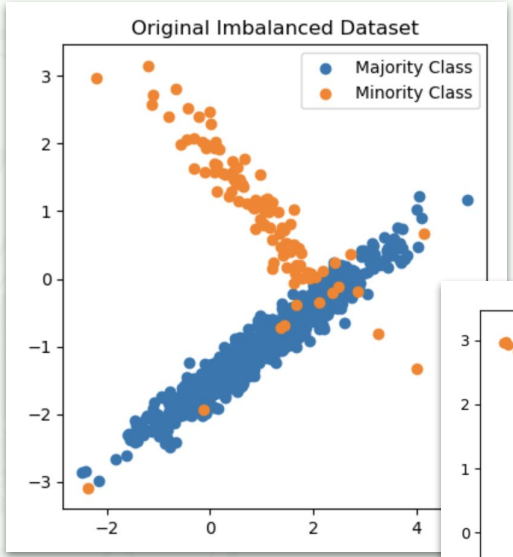


1. Select a minority class instance
2. Find its k-nearest neighbors
3. Choose one of the neighbors randomly
4. Create a new synthetic instance along the line between the two points

SMOTE: **S**ynthetic **M**inority **O**ver-Sampling **Te**chnique



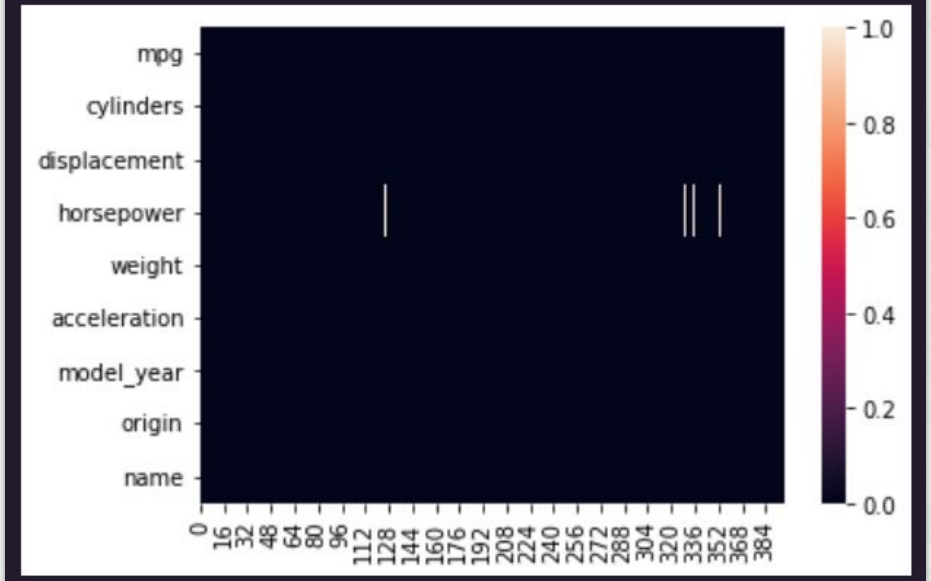
Python Library



Missing Values

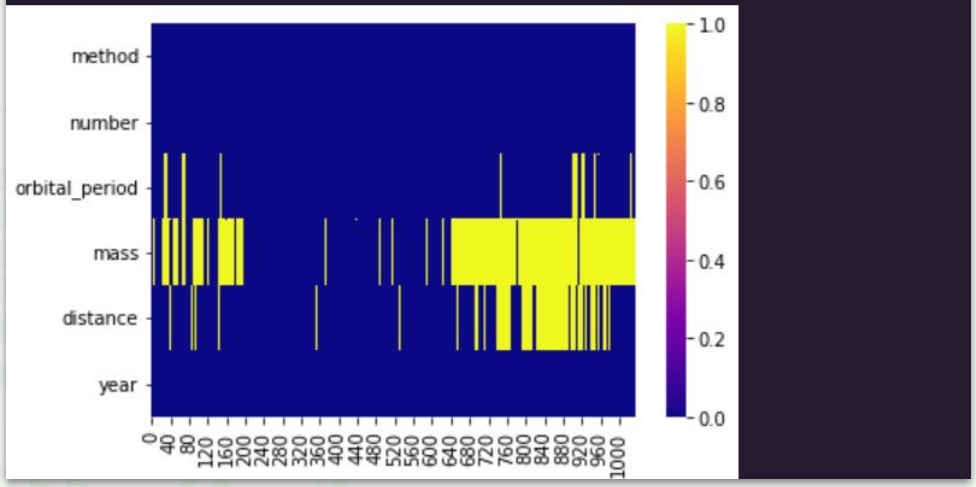
Visualize Missing Data: Part of the EDA Process

```
1 sns.heatmap(df_m.isna().transpose());
```



Seaborn: mpg

```
1 sns.heatmap(df_pl.isna().transpose(),cmap="plasma");
```



Seaborn: planets

Handling Missing Values: Simple Approach

Suppose you don't care *why* you have missing values. Use *scikit-learn*'s `SimpleImputer`.

API Reference > sklearn.impute > SimpleImputer

SimpleImputer

```
class sklearn.impute.SimpleImputer(*, missing_values=nan, strategy='mean',
fill_value=None, copy=True, add_indicator=False, keep_empty_features=False)
```

Univariate imputer for completing missing values with simple strategies. [source]

Replace missing values using a descriptive statistic (e.g. mean, median, or most frequent) along each column, or using a constant value.

strategy : *str or Callable, default='mean'*

The imputation strategy.

- If "mean", then replace missing values using the mean along each column. Can only be used with numeric data.
- If "median", then replace missing values using the median along each column. Can only be used with numeric data.
- If "most_frequent", then replace missing using the most frequent value along each column. Can be used with strings or numeric data. If there is more than one such value, only the smallest is returned.
- If "constant", then replace missing values with fill_value. Can be used with strings or numeric data.
- If an instance of Callable, then replace missing values using the scalar statistic returned by running the callable over a dense 1d array containing non-missing values of each column.

Missingness: Quality/Mechanism of Being Missing

In data science, missingness refers to the absence or unavailability of data in a dataset. It's a concept that deals with how to handle and interpret missing values within a dataset. There are generally three main types or qualities of missingness:

- **Missing Completely at Random (MCAR)**: The probability of data being missing is the same for all observations. The missingness is unrelated to any variables in the dataset.
- **Missing at Random (MAR)**: The probability of missing data depends on observed data, but not on the missing data itself.
- **Missing Not at Random (MNAR)**: The probability of missing data depends on unobserved data, including the missing data itself.

Understanding the type of missingness in a dataset is crucial for choosing appropriate strategies to handle missing data, such as imputation methods or adjusting analysis techniques.

Missingness in Terms of Conditional Probabilities

MCAR:

$$P(\text{missing}|\text{complete}) = P(\text{missing})$$

MAR:

$$P(\text{missing}|\text{complete}) = P(\text{missing}|\text{observed})$$

MNAR:

$$P(\text{missing}|\text{complete}) \neq P(\text{missing}|\text{observed})$$

Missingness Diagrams

Let *R* be a binary variable such that *R=1* indicates missing and *R=0* indicates observed.

W is the complete dataset.

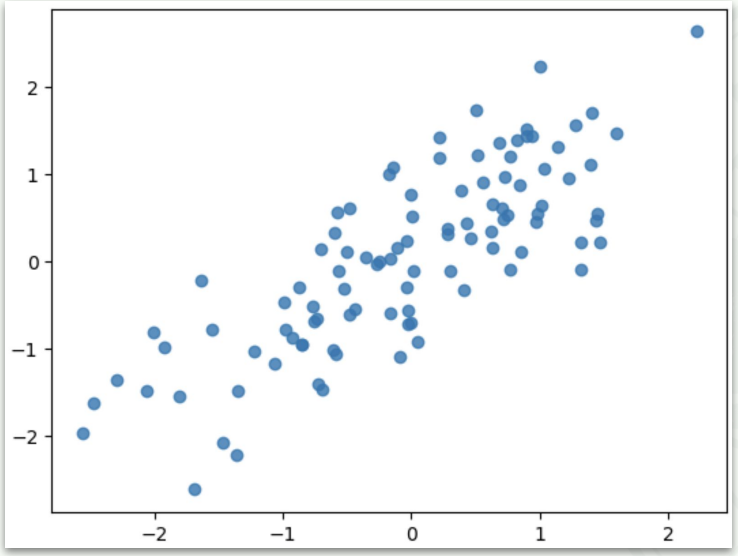
X_{obs} is the observed portion of the data.

X_{mis} is the missing portion of the data.

Z is the variable we are concerned with.

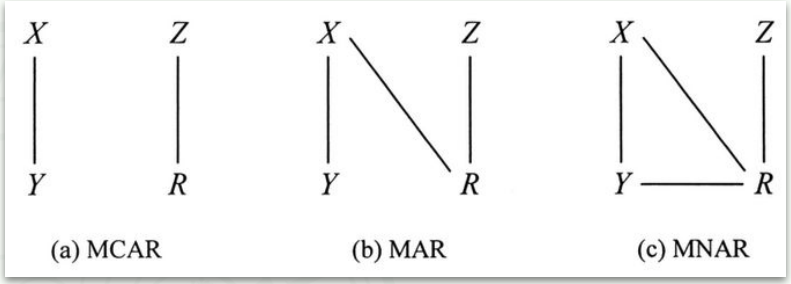
X and *Y* are some other variable, fully observed. *W = {X,Y,Z}*

We Can Create The Types of Missingness



original

Z is the cause
X is observed
Y is partly missing



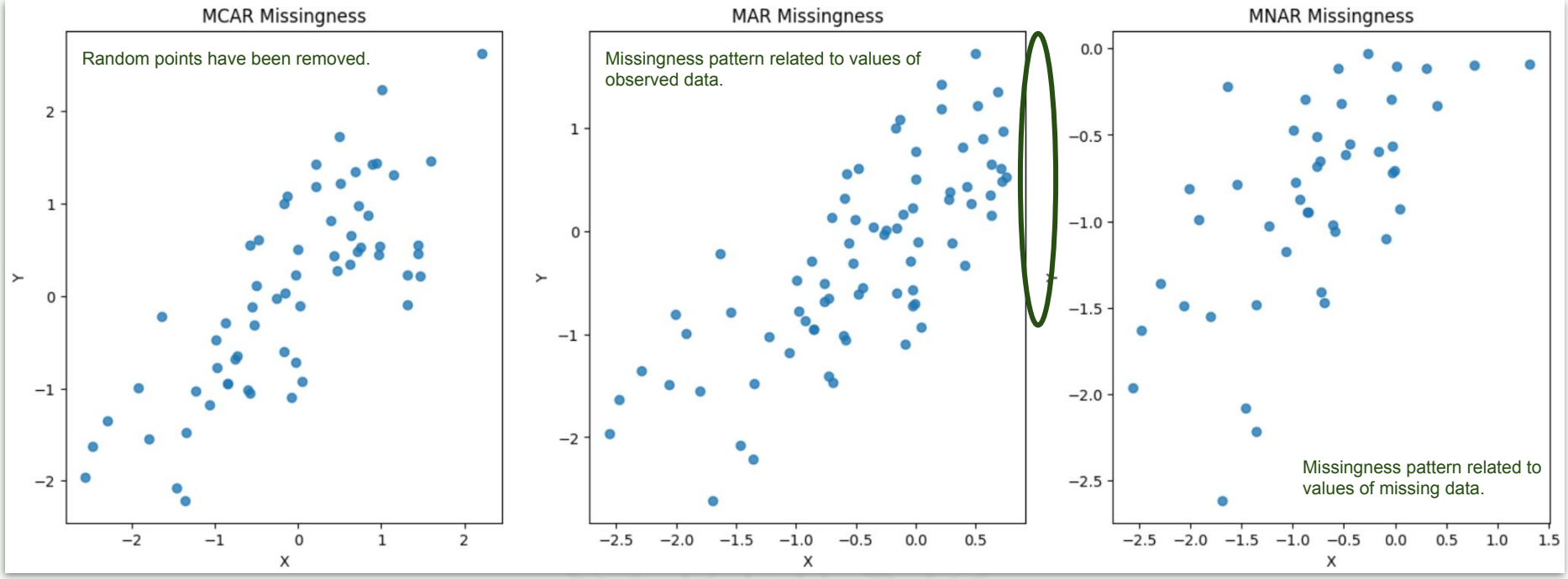

```python
# Introduce MCAR missingness
missing_rate = 0.2  # Proportion of missing values
df_mcar = df_original.copy()
for col in df_mcar.columns:
    missing_indices = random.sample(range(n), int(missing_rate * n))
    df_mcar.loc[missing_indices, col] = np.nan

# Introduce MAR missingness
df_mar = df_original.copy()
missing_indices_mar = df_original[df_original['X'] > df_original['X'].quantile(0.75)].index
df_mar.loc[missing_indices_mar, 'Y'] = np.nan

# Introduce MNAR missingness
df_mnar = df_original.copy()
missing_indices_mnar = df_original[df_original['Y'] > 0].index
df_mnar.loc[missing_indices_mnar, 'Y'] = np.nan
```

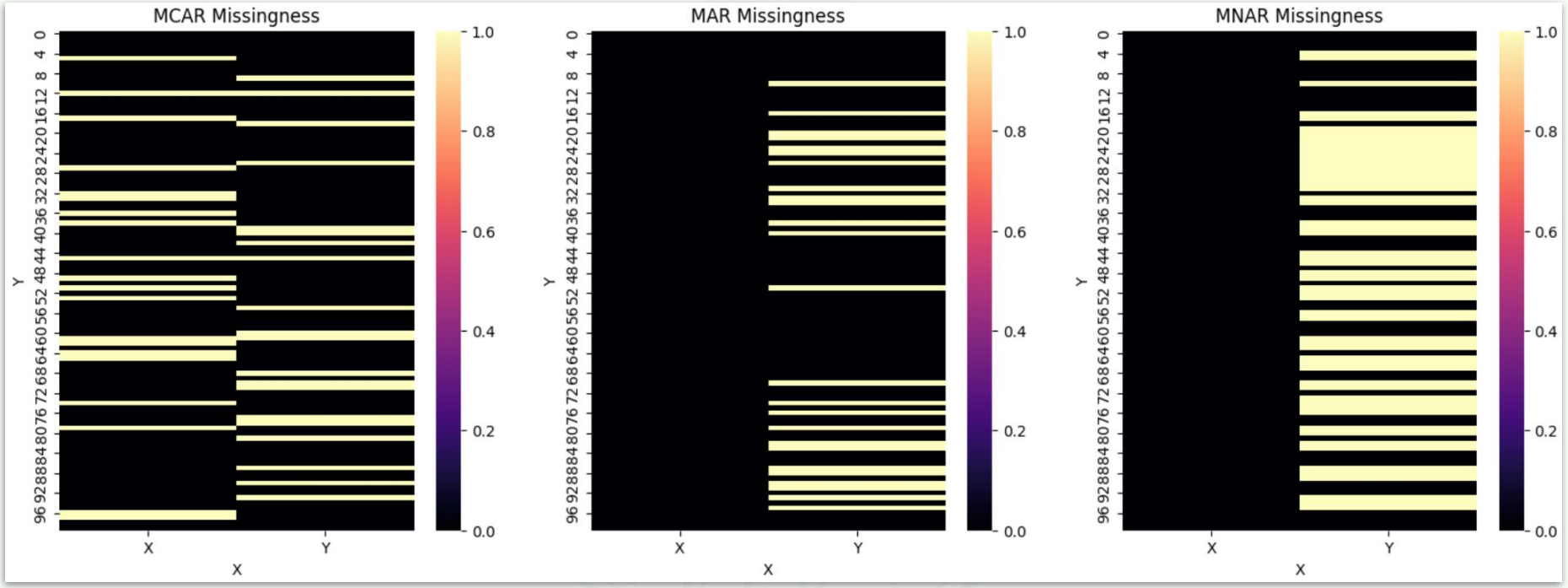
add missingness to the data.

What Do The Types of Missingness Look Like?

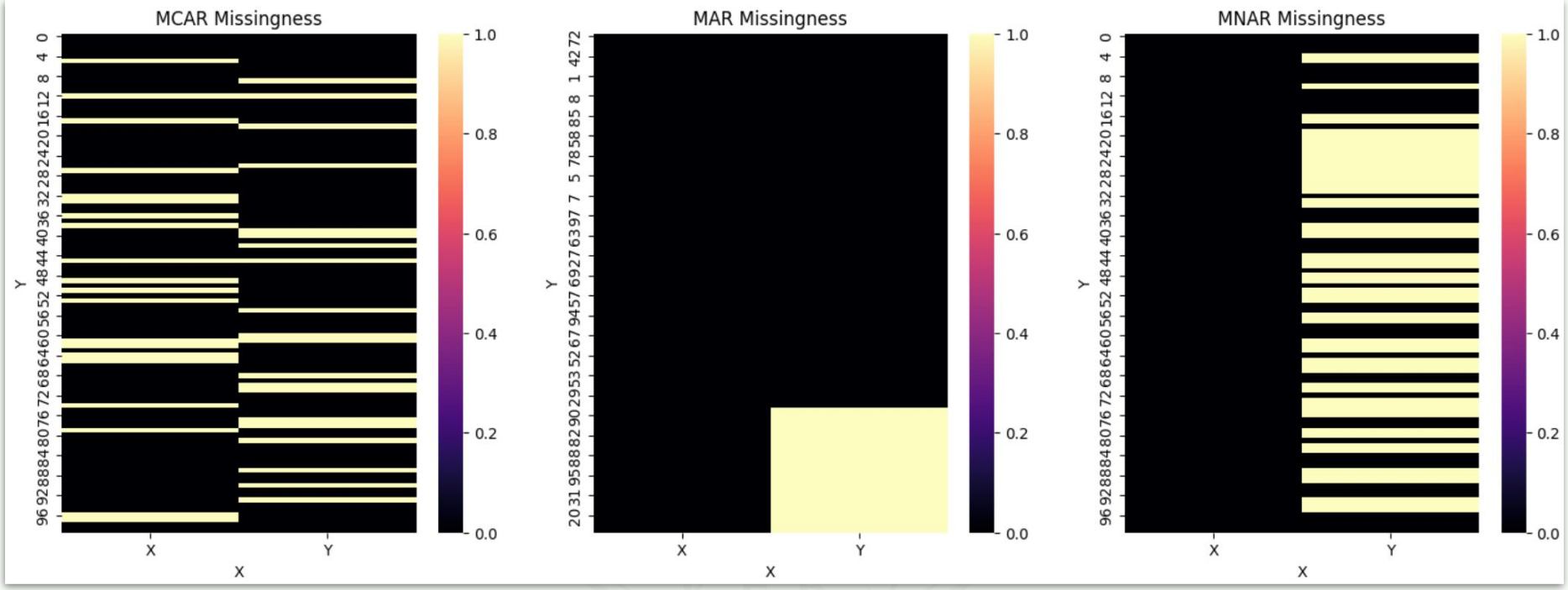


Note the range of the axes.

Visualizing Synthetic Missingness



Sorted by X Value



No pattern emerges.

Missingness related to observed values.

Missingness related to unobserved values.

Worse Case Scenario

Your data could be MCAR, MAR and MNAR at the same time.

Missingness is difficult to diagnose

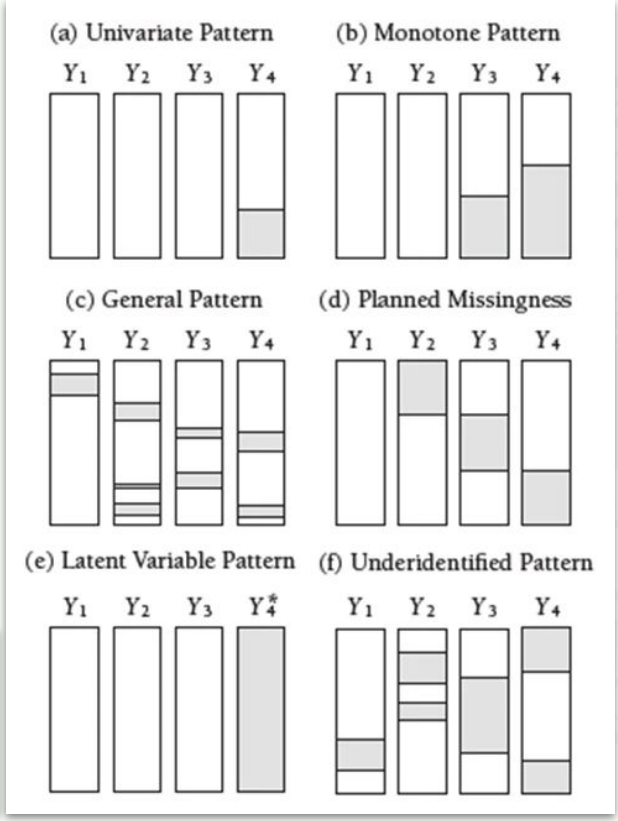
Correlation Analysis: Test for Missingness

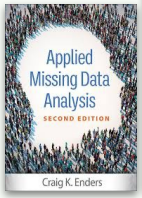
1. create a new column in the data matrix: *N*
2. place a 0 or 1 in that new column to indicate missing or not in *Y*
3. treat the column *X* and the new column *N* as a machine learning problem: can *X* predict *N*?
4. if so, the data is partially MAR

When? Why?

Area	Sub-categories	Description	Examples
Types of Missing Data Values	1. Structural missing values 2. Intermittent missing values 3. Dropouts	Different ways in which data can be missing in a dataset	1. Questions not applicable to all respondents 2. Skipped questions in a survey 3. Participants leaving a longitudinal study
Causes of Missingness	1. Data entry errors 2. Equipment malfunction 3. Participant non-response 4. Study design	Reasons why data might be missing	1. Typos or omissions during data input 2. Sensor failure in IoT data collection 3. Refusal to answer sensitive questions 4. Planned missing data designs
Patterns of Missingness	1. Univariate 2. Monotone 3. Arbitrary	How missing data is distributed across variables	1. Missing data in only one variable 2. If a variable is missing, all subsequent variables are missing 3. Any variable can be missing for any observation
Impact of Missingness	1. Bias in parameter estimates 2. Loss of statistical power 3. Complications in data analysis	How missing data affects statistical analyses and results	1. Underestimation of population parameters 2. Increased Type II error rate 3. Need for specialized analytical techniques

Missingness Patterns





Pattern	Description	Example	Potential Mechanism	Challenges	Handling Strategies
Univariate	Missing data occurs in only one variable	Income data missing for some participants	Can be MCAR, MAR, or MNAR	Relatively simple to address	Single imputation, multiple imputation, or complete case analysis depending on mechanism
Monotone	If a variable is missing, all subsequent variables are also missing	Longitudinal study where participants drop out over time	Often MAR, but can be MCAR or MNAR	Common in longitudinal data	Special imputation methods for monotone patterns, mixed-effects models
General	No clear pattern to the missing data	Multiple variables have missing values with no apparent structure	Can be MCAR, MAR, or MNAR	Most complex to handle	Multiple imputation, maximum likelihood methods
Planned	Missingness is deliberately introduced into the study design	Matrix sampling in surveys, where different subsets of questions are given to different respondents	Usually MCAR if properly designed	Can reduce respondent burden but increases complexity of analysis	Planned missing data designs, multiple imputation
Latent	Missingness is related to an unmeasured (latent) variable	Depression affects both survey participation and responses	Often leads to MNAR	Difficult to detect and address	Sensitivity analysis, selection models, pattern-mixture models
Underidentified	Not enough observed data to estimate missing data parameters	Too many variables with missing data relative to observed data	Can occur with any mechanism	Makes it impossible to fully recover missing information	Requires strong assumptions or additional data sources

Remember....

Patterns and qualities are not mutually exclusive.

They almost always happen in combination.

Missing data analysis requires a nuanced approach!

Planned Missingness

Reason	Description	Benefit
1. Reduce Respondent Burden	Shorten individual surveys by distributing questions across participants	Decreases fatigue, improves response quality, reduces dropout rates
2. Cost Efficiency	Collect expensive data from only a subset of participants	Reduces overall study costs while still obtaining valuable data
3. Expand Scope of Study	Include more variables without making surveys excessively long	Allows addressing a broader range of research questions within a single study
4. Improve Data Quality	Shorter surveys yield higher quality responses	Increases accuracy and reliability of collected data
5. Increase Response Rates	Shorter surveys are more likely to be completed	Potentially improves overall participation and representativeness of the sample
6. Enable Methodological Comparisons	Compare results from complete data to imputed data	Provides insights into imputation methods and their effectiveness
7. Address Ethical Concerns	Avoid asking all questions to all participants when inappropriate	Allows inclusion of sensitive topics or time-consuming assessments without burdening all participants