PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY of WASHINGTON

Introduction to Machine Learning

MLEARN 510A – Lesson 4

Mid-Course Survey Results

- 60% response so far! Please fill it if you haven’t already
- Mathematical aspects are useful
- Supplementary material (quizzes, exercises, case studies) is useful
- Go slow and spend more time on mathematical details
- Feedback on assignments/Early posting of assignments

Recap of Lesson 3

- Introduction to Classification
- Logistic Regression and Maximum Likelihood
- Logistic Regression Extensions
- Linear Discriminant Analysis (LDA)
- Quadratic Discriminant Analysis (QDA)
- Comparison of Various Algorithms

Outline for Lesson 4

- Data Preprocessing
- Dealing with Missing Data
- Detection of Outliers
- Exploratory Data Analysis
- Data Transformations
- Data Splitting

Data Preprocessing

- **Data preprocessing** is an important step in the data mining process

- Includes
  - Data cleaning
  - Data transformation
  - Feature extraction

- Product of data preprocessing is the final training set

- Requires experience and practice!

Why Data Preprocessing?

Garbage in ⟶ Garbage out

Image from: Dilbert.com

- Data in real world is “dirty”:
  - Incomplete
  - Noisy
  - Inconsistent

- Without quality data, there are no quality models

- Data quality is described in terms of:
  - Accuracy
  - Completeness
  - Conformity
  - Consistency
  - Integrity
  - Timeliness

Major Tasks in Data Preprocessing

- Data cleaning
  - Fill in missing values, smooth noisy data, identify & remove outliers, resolve inconsistencies
- Data integration
  - Integrate with other sources of data
- Data transformation
  - Normalize and aggregate
- Data reduction
  - Reduce volume, while keeping most of the information

Missing Values in Data

- Defined as the data value that is not stored for a variable in the observation of interest
- Common in almost all research and can have a significant effect on the conclusions that can be drawn from the data
- Missing data
  - Reduces statistical power of a test
  - Can cause bias in the estimation of parameters
  - Complicates later analysis

Types of Missing Values

- **Missing Completely at Random (MCAR)**
  - The probability that a variable value is missing does not depend on the observed data values nor on the missing data values
- **Missing at Random (MAR)**
  - The probability that a variable value is missing partly depends on other observed data, but does not depend on any of the values that are missing
- **Missing Not at Random (MNAR)**
  - The probability that a variable value is missing depends on the missing data values themselves

An Example

- Imagine two variables X and Y, where some of the data on Y are missing
- Now imagine a dummy variable miss(y), which is coded as 0 when Y is observed and coded as 1 when Y is missing
- **MCAR:** miss(y) is not related to Y or to X
- **MAR:** miss(y) is related to X (i.e., one can predict whether Y is missing based on observed values of X), but miss(y) is not related to Y after X is controlled
- **MNAR:** miss(y) is related to Y itself (i.e., related to the missing values of Y), even after X is controlled

Techniques for Handling Missing Data

Listwise Deletion:

- Removes all data for a case that has one or more missing values
- W/O MCAR, it is biased

Pairwise Deletion:

- Maximizes all data available by retaining data which is required for an analysis

Variable Deletion:

- Discard variable which is missing values

Listwise Deletion

PID	Survived	Pclass	Age	SibSp	Parch	Ticket	Fare	Cabin	Emb.
~~1~~	~~0~~	~~3~~	~~22~~	~~1~~	~~0~~	~~A/5 21171~~	~~7.25~~	~~X~~	~~S~~
2	1	1	38	1	0	PC 17599	71.2833	C85	C
~~3~~	~~1~~	~~3~~	~~26~~	~~0~~	~~0~~	~~STON/O2.~~	~~7.925~~	~~X~~	~~S~~
4	1	1	35	1	0	113803	53.1	C123	S
~~5~~	~~0~~	~~3~~	~~35~~	~~0~~	~~0~~	~~373450~~	~~8.05~~	~~X~~	~~S~~
~~6~~	~~0~~	~~3~~	~~X~~	~~0~~	~~0~~	~~330877~~	~~8.4583~~	~~X~~	~~Q~~

Pairwise / Variable Deletion

PID	Survived	Pclass	Age	SibSp	Parch	Ticket	Fare	Cabin	Emb.
1	0	3	22	1	0	A/5 21171	7.25	X	S
2	1	1	38	1	0	PC 17599	71.2833	C85	C
3	1	3	26	0	0	STON/O2.	7.925	X	S
4	1	1	35	1	0	113803	53.1	C123	S
5	0	3	35	0	0	373450	8.05	X	S
6	0	3	X	0	0	330877	8.4583	X	Q

Missing Data Techniques – Single Imputation

- Substitute missing value with:
  - Single value (e.g. mean, median, worst case, best case)
  - Values dynamically from the dataset (e.g. nearest value)

PID	Survived	Pclass	Age	SibSp	Parch	Ticket	Fare	Cabin	Emb.
1	0	3	22	1	0	A/5 21171	7.25	X	S
2	1	1	38	1	0	PC 17599	71.2833	C85	C
3	1	3	26	0	0	STON/O2.	7.925	X	S
4	1	1	35	1	0	113803	53.1	C123	S
5	0	3	35	0	0	373450	8.05	C	S
6	0	3	33	0	0	330877	8.4583	D	Q

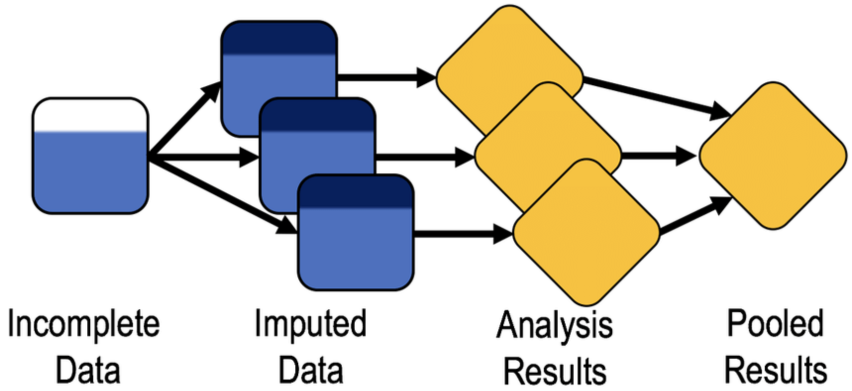
- Single value (especially mean) is often a bad estimate

Missing Data Techniques – Multiple Imputation

- **Imputation**: Create *n* sets of imputations for the missing values

- **Analysis**: Use standard statistical methods to fit the model of interest to each of the imputed datasets

- **Pooling**: Combine results, calculating the variation in parameter estimates.



Missing Data Techniques

	Missingness Mechanism		
Missing Data Technique	MCAR	MAR	MNAR
Listwise Deletion	Unbiased; Large Std. Errors (Low Power)	Biased; Large Std. Errors (Low Power)	Biased; Large Std. Errors (Low Power)
Pairwise Deletion	Unbiased; Inaccurate Std. Errors	Biased; Inaccurate Std. Errors	Biased; Inaccurate Std. Errors
Single Imputation	Often Biased; Inaccurate Std. Errors	Often Biased; Inaccurate Std. Errors	Biased; Inaccurate Std. Errors
Maximum Likelihood (ML)	**Unbiased; Accurate Std. Errors**	**Unbiased; Accurate Std. Errors**	Biased; Accurate Std. Errors
Multiple Imputation (MI)	**Unbiased; Accurate Std. Errors**	**Unbiased; Accurate Std. Errors**	Biased; Accurate Std. Errors

Note. Recommended techniques are in boldface. Adapted from Newman (2009).

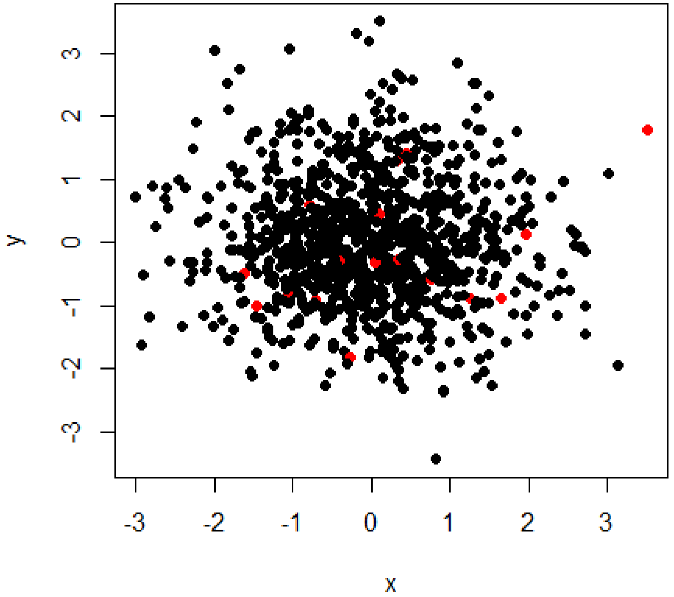
Newman, Missing Data: Five Practical Guidelines, Organizational Research Methods 2014

Dealing With Outliers

- Handling outliers is dependent on the nature of data
- If a small fraction of data points are outliers, we can consider dropping those data points
- In a different scenario, outliers might be invaluable signals for one of the classes
- Choose algorithms that robust to outliers

Dealing With Class Imbalance

- ➢ Collect more data
- ➢ Try resampling your dataset
- ➢ Generate synthetic samples (SMOTE)
- ➢ Change performance metric
- ➢ Use a different algorithm
- ➢ Use penalized models



Exploratory Data Analysis (EDA)

- Used by data scientists to analyze and investigate data sets
- Often the first step in an ML project
- Makes it easier to discover patterns, spot anomalies, test a hypothesis, or check assumptions
- Enables preliminary selection of appropriate models
- Complements inferential statistics

Types of EDA

- First categorization is whether the method is graphical or non-graphical
- Second level is whether analysis is univariate or multivariate
- Therefore, we have four types of EDA
  - Univariate non-graphical
  - Multivariate non-graphical
  - Univariate graphical
  - Multivariate graphical

Univariate Non-graphical EDA

- Simplest form of data analysis, where the data being analyzed consists of just one variable
- Goal is to better appreciate the “sample distribution”
- Make some tentative conclusions about what population distribution(s) is/are compatible with the sample distribution
- Outlier detection is also a part of this analysis
- Different methods for categorical and continuous variables

Univariate Non-graphical EDA – Categorical Variables

- Look at the range of values and the frequency (or relative frequency) of occurrence for each value

- **A simple tabulation of the frequency of each category is the best univariate non-graphical EDA for categorical data**

Statistic/College	H&SS	MCS	SCS	other	Total
Count	5	6	4	5	20
Proportion	0.25	0.30	0.20	0.25	1.00
Percent	25%	30%	20%	25%	100%

Univariate Non-graphical EDA – Continuous Variables

- Make preliminary assessments about the population distribution of the variable using the data of the observed sample

- Our observed data represent just one sample out of an infinite number of possible samples

- Look for ‘sample statistics’
  - Sample mean
  - Sample variance
  - Sample standard deviation
  - Sample skewness and
  - Sample Kurtosis

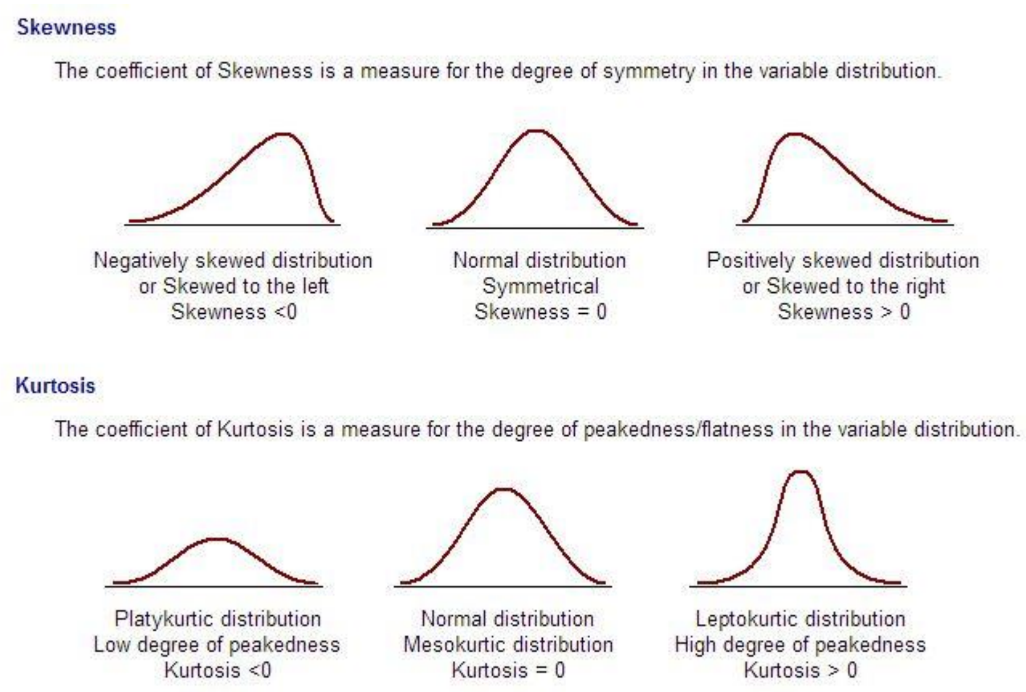
Univariate Non-graphical EDA – Continuous Variables

- **Central tendency** or location" of a distribution has to do with typical or middle values
  - Mean: This is the most often used measure
  - Median: Robust to outliers. Used when the distribution is skewed
  - Mode: Most frequent value. Not used very often

- **Spread** is an indicator of how far away from the center we are still likely to find data values
  - Standard deviation/Variance
  - Quantiles
  - Inter-quartile range (IQR) → IQR = Q3 – Q1. More robust measure than variance

Other Measure of Spread

- **Skewness**: a measure of asymmetry
- **Kurtosis**: a measure of “peakedness" relative to a Gaussian shape



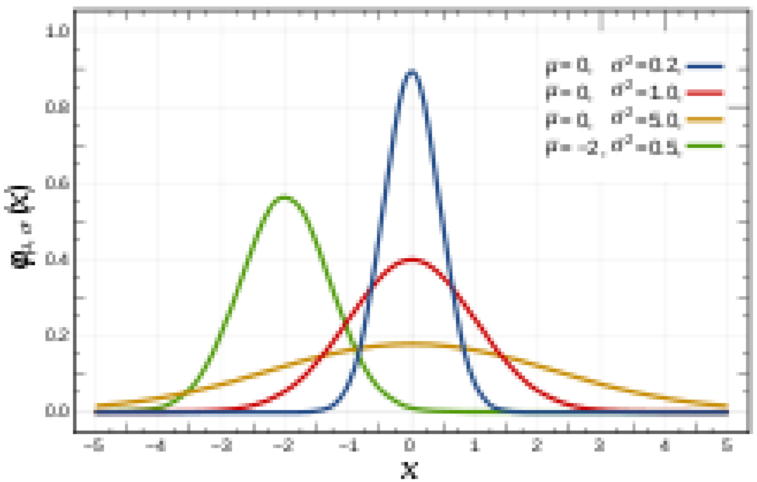
Quiz

- *Can you think of a distribution for which the mean, median and mode are all equal? Give at least two examplees*

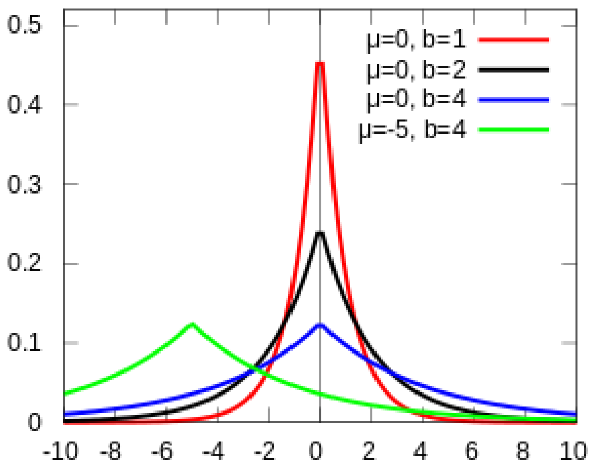
Quiz

➢ *Can you think of a distribution for which the mean, median and mode are all equal?*

Gaussian Distribution

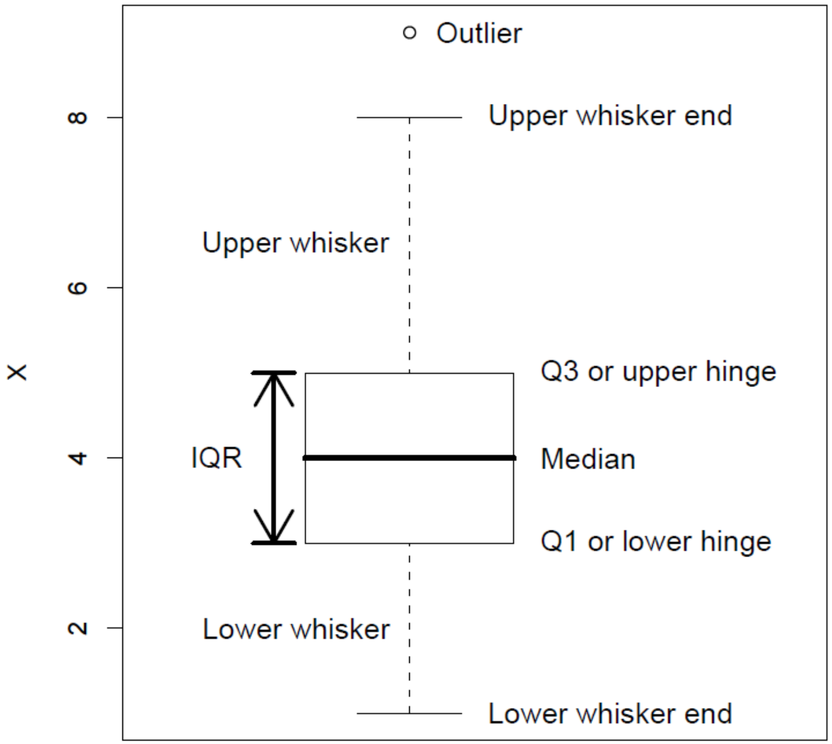


Laplace Distribution



Univariate Graphical EDA

- **Histograms**: show central tendency, spread, modality, shape and outliers
- **Boxplots**: show robust measures of location and spread as well as providing information about symmetry and outliers



Multivariate Non-graphical EDA

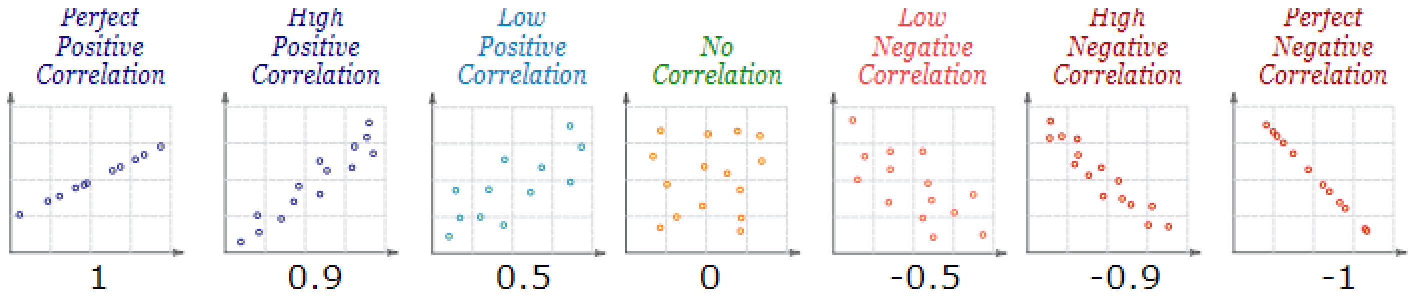
- Shows the relationship between two or more variables in the form of either cross-tabulation or statistics

- **Cross-tabulation**: the basic bivariate non-graphical EDA technique
  - Making a two-way table with column headings that match the levels of one variable and row headings that match the levels of the other variable
  - Fill in the counts of all subjects that share a pair of levels

Age Group / Sex	Female	Male	Total
young	2	3	5
middle	2	1	3
old	3	0	3
Total	7	4	11

Multivariate Non-graphical EDA

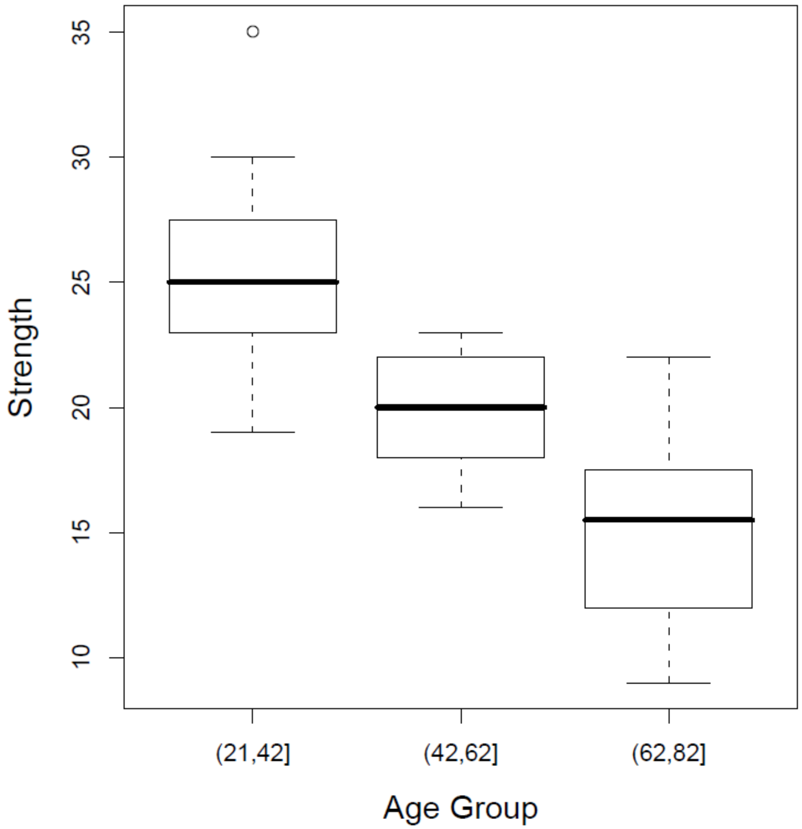
- Correlation



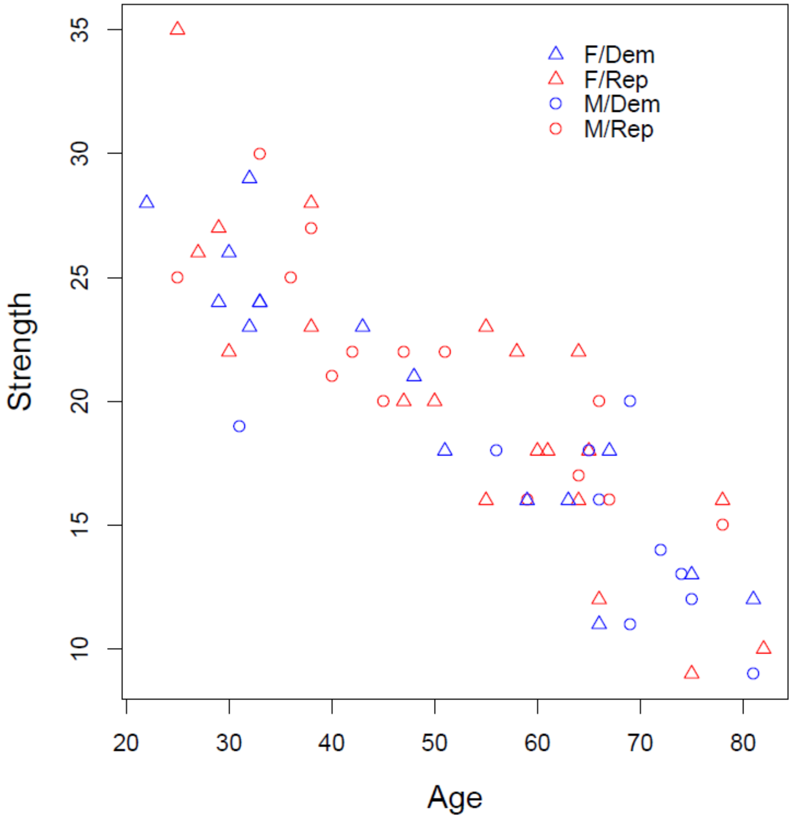
- Covariance
  - Positive covariance → when one measurement is above the mean the other will probably also be above the mean
  - Negative covariances → when one variable is above its mean, the other is below its mean

Multivariate Graphical EDA

Side-by-side box plots



Scatter plots



Wrapping Up EDA

- Perform appropriate EDA before further analysis of your data
- Perform whatever steps are necessary to become more familiar with your data
- Check for obvious mistakes, learn about variable distributions, and learn about relationships between variables
- EDA is not an exact science – it is a very important art!
- Get better with practice and experience

Quiz

- Which graphical method do you prefer for EDA?
- Which non-graphical method do you prefer for EDA?

Data Transformation

- Data transformation processes transform raw variables into meaningful variables
- Relevant
  - Provide useful information to discriminate between categories
- Discriminative
  - There is enough variability between training examples of different classes
- Non-redundant
  - Unlike already developed features

Feature Engineering

- Feature transformation
  - Transforming existing feature into one with a specific function
- Feature construction
  - Turning raw data into informative features that algorithm can understand
- Dimensionality reduction
  - Reduce number of features, while preserving overall information content

Encoding Categorical Target Variable

- Categorical target variables need to be transformed to numerical values for use in ML models

- **Label Encoding:** normalize labels such that they contain only values between 0 and n_classes-1

```
from sklearn.preprocessing import LabelEncoder
```

- If used with feature variables, it introduces ordinal structure which may not be suitable

Encoding Categorical Features

- Categorical variables need to be transformed to numerical values for use in ML models

- **One-hot Encoding:** features are encoded using a one-hot (aka ‘one-of-K’ or ‘dummy’) encoding scheme. This creates a binary column for each category

```
from sklearn.preprocessing import OneHotEncoder
```

- Can quickly grow in size if variable takes on many unique values

One-hot vs. Label Encoding

- **Label Encoding:** normalize labels such that they contain only values between 0 and n_classes-1

- **One-hot Encoding:** features are encoded using a one-hot (aka 'one-of-K' or 'dummy') encoding scheme. This creates a binary column for each category

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

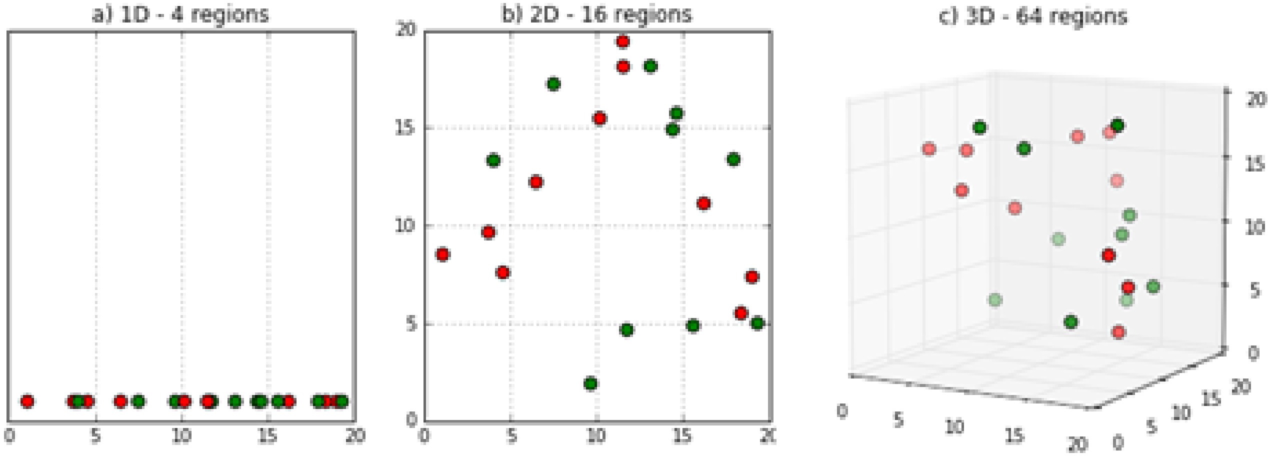
Quiz

- *Can you think of a feature that should not be encoded using one-hot encoding?*

- *Can you think of a feature that can be modeled using label encoding?*

Curse of Dimensionality

- Using one-hot encoding of such feature might lead to *Curse of Dimensionality*
- If we have more features than observations than **we run the risk of massively overfitting our model — this would generally result in terrible out of sample performance**



Handling High Cardinality Features

- ***Supervised Ratio***

$v_i = p_i / t_i$ *where*
v_i = *numerical value for* i^{th} *value of some categorical attribute*
p_i = *number of records with positive class value for the categorical attribute value in question*
t_i = *total number of records with the categorical attribute value in question*

- ***Weight of Evidence***

$v_i = log((p_i / p) / (n_i / n)$ *where*
p_i = *number of records with positive class value for the categorical attribute value in question*
n_i = *number of records with negative class value for the categorical attribute value in question*
p = *total number of records with positive class value*
n = *total number of records with negative class value*

Discretization

- Partitions continuous features into discrete values

- Certain datasets with continuous features may benefit from discretization, because discretization can transform the dataset of continuous attributes to one with only nominal attributes

```
>>> X = np.array([[ -3., 5., 15 ],
...               [  0., 6., 14 ],
...               [  6., 3., 11 ]])
>>> est = preprocessing.KBinsDiscretizer(n_bins=[3, 2, 2], encode='ordinal').fit(X)
```

feature 1: $[-\infty, -1), [-1, 2), [2, \infty)$

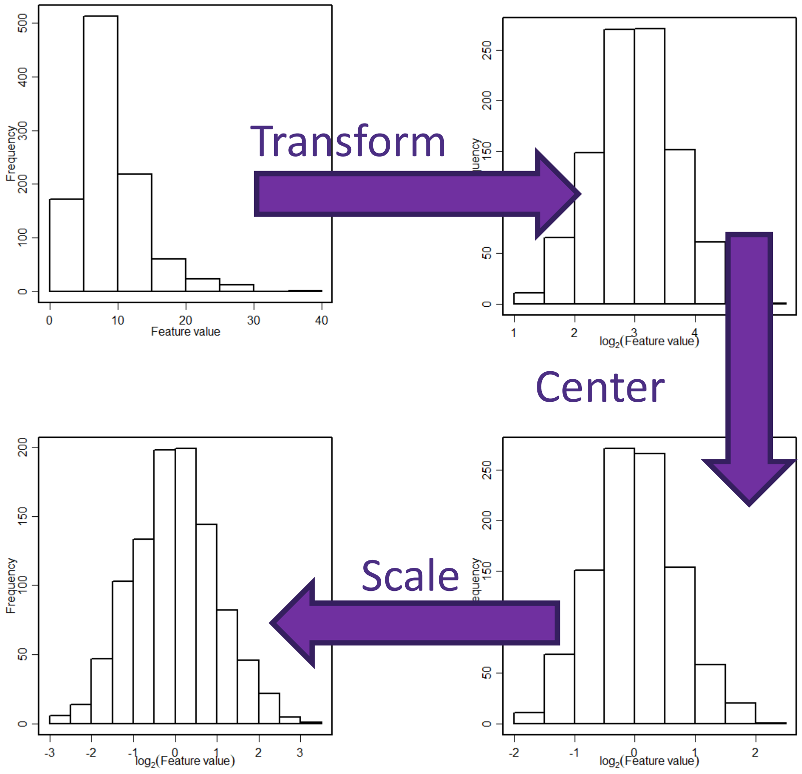
feature 2: $[-\infty, 5), [5, \infty)$

feature 3: $[-\infty, 14), [14, \infty)$

```
>>> est.transform(X)
array([[ 0., 1., 1.],
       [ 1., 1., 1.],
       [ 2., 0., 0.]])
```

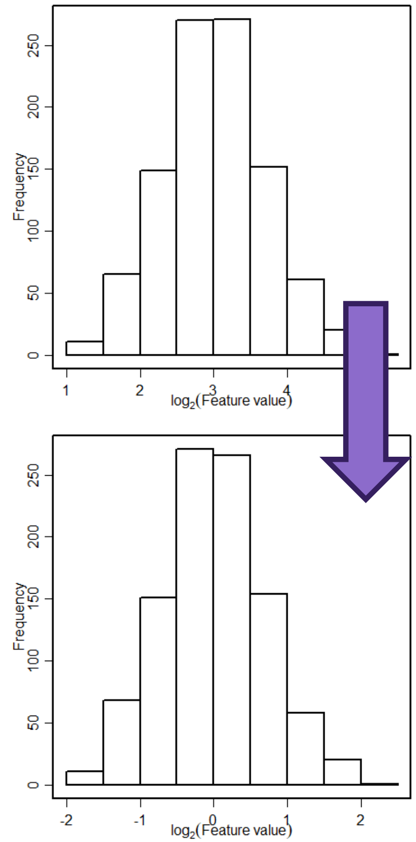
Correcting Distributions

- Some modelling techniques (e.g. LR) perform better with variables that have been monotonically transformed
- These transformations are divided into:
  - Centering
  - Scaling
  - Transformation



Centering

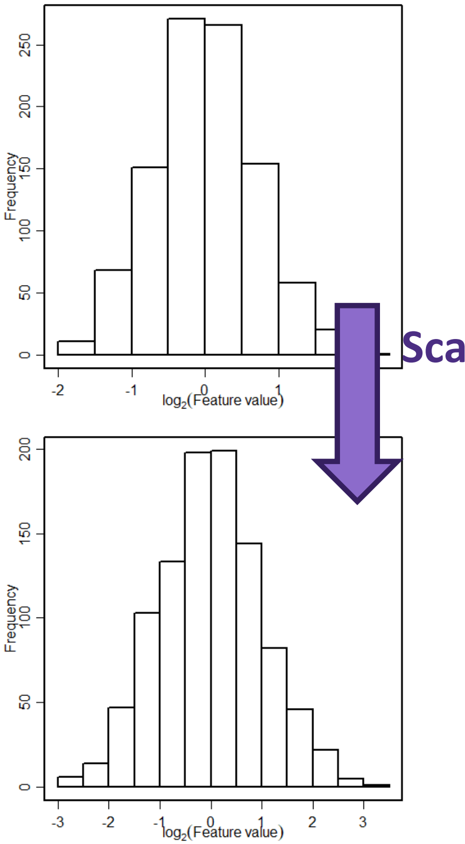
- Shift the 'center' of the feature to 0
- Center can be defined as:
  - Mean
  - Median
  - Mode



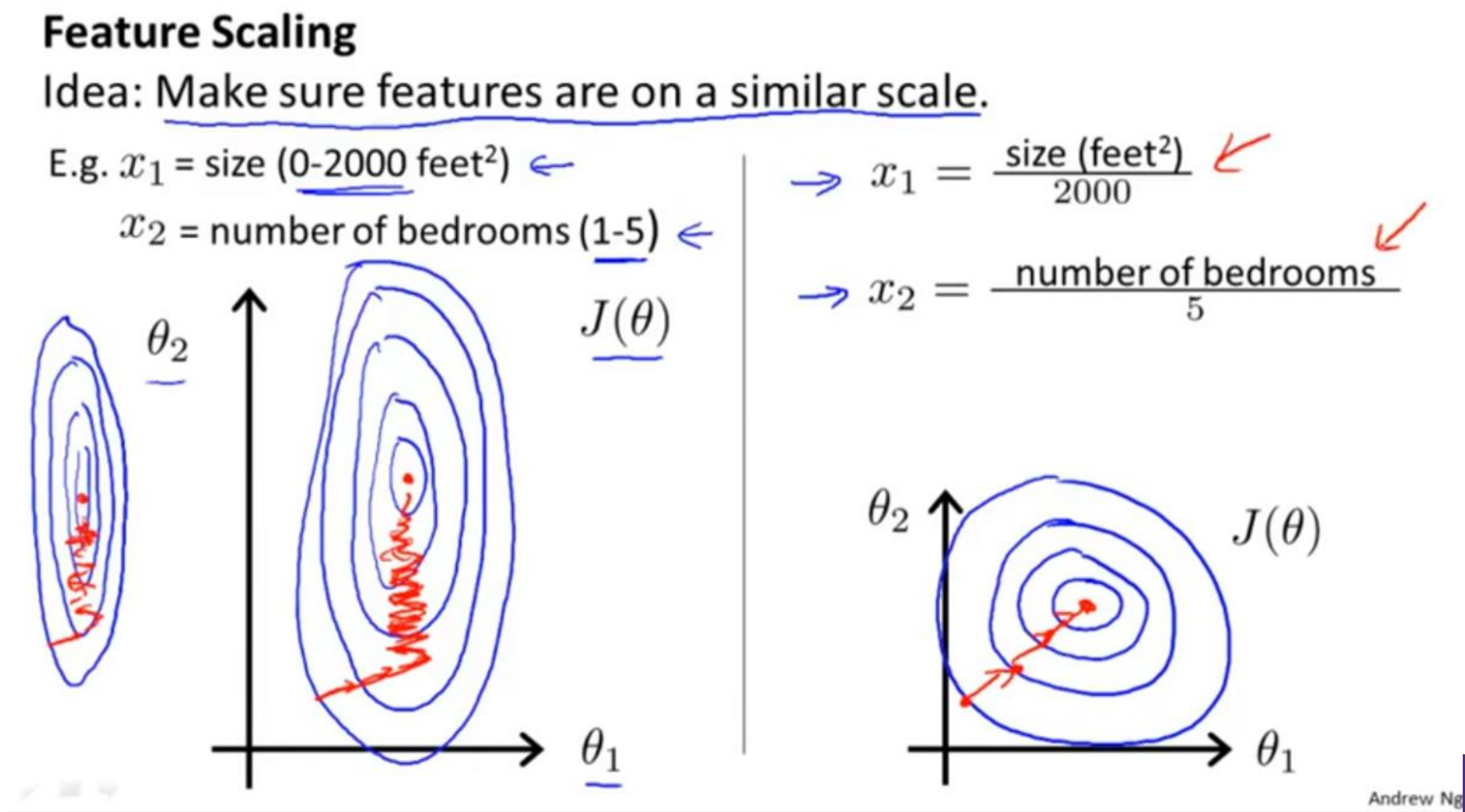
Scaling

- Methods:
  - Standard (Z) scaling
  - Min-Max (range) scaling
  - Pareto scaling
  - Vast scaling
  - Level scaling

- Scaling is important only for some of the algorithms:
  - K-means, K-NN, if you want all features to contribute equally to prediction
  - Regression methods, SVMs, perceptrons, neural networks, to improve performance of gradient-descent based optimizers
  - LDA, PCA, to ensure that all features contribute equally



Effect of Scaling on Parameter Optimization



Transformations

Types:

- Box-cox transform

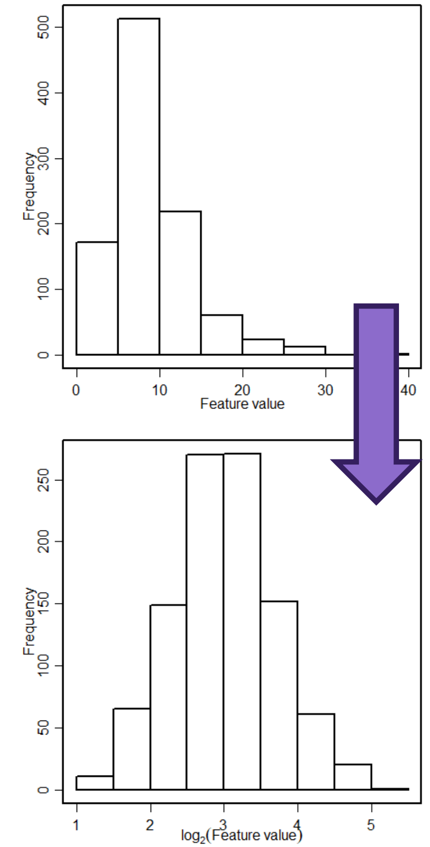
$$x^* = \begin{cases} \frac{x^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \log(x) & \text{if } \lambda = 0 \end{cases}$$

- Log transforms

$$\tilde{x}_{ij} = {}^{10}\log(x_{ij})$$
$$\widehat{x}_{ij} = \tilde{x}_{ij} - \bar{\tilde{x}}_i$$

- Power transforms

$$\tilde{x}_{ij} = \sqrt{(x_{ij})}$$
$$\widehat{x}_{ij} = \tilde{x}_{ij} - \bar{\tilde{x}}_i$$



Reasons:

- Make distributions more normal-like (or symmetric)
- Reduce heteroscedasticity
- Convert multiplicative relationships to additive ones

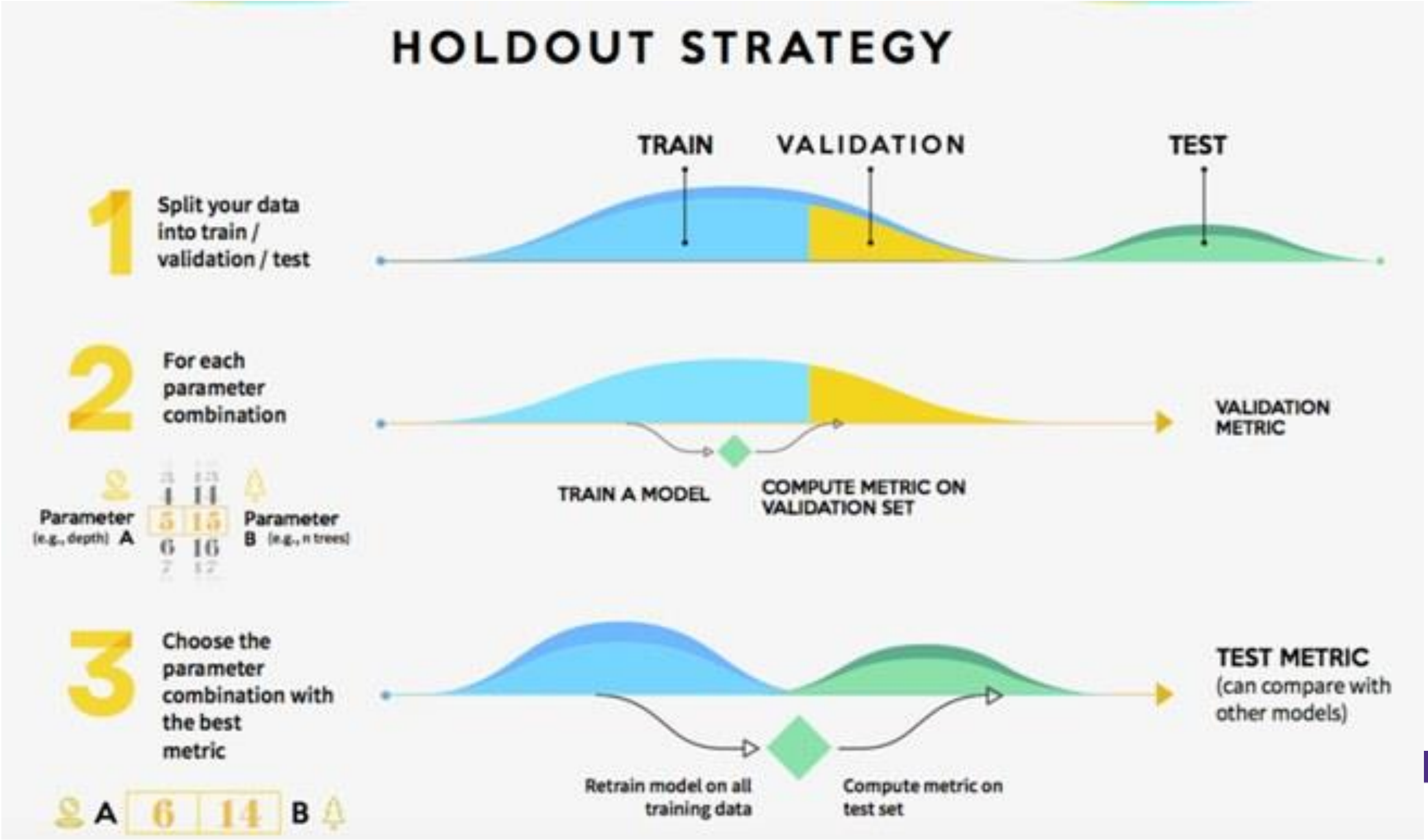
Exercise

- Check out ***sklearn.preprocessing*** for various preprocessing techniques

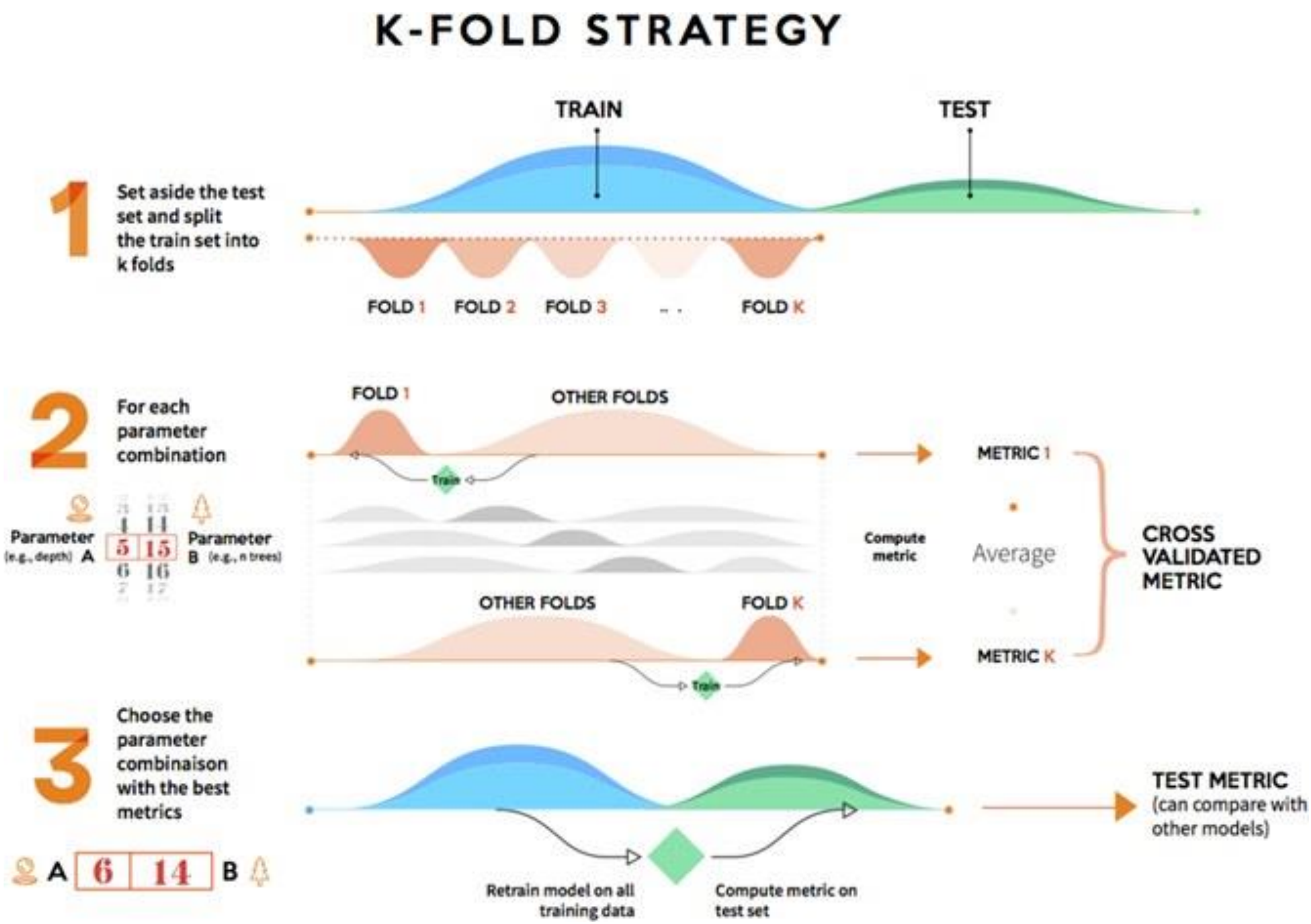
Data Splitting in ML

- The fundamental goal of ML is to *generalize* beyond the data instances used to train models
- Future instances have unknown target values, and we cannot check the accuracy of our predictions for future instances now
- Need to use some of the data that we already know the answer for as a proxy for future data
- Carve out a ‘test’ dataset

Hold Out Strategy

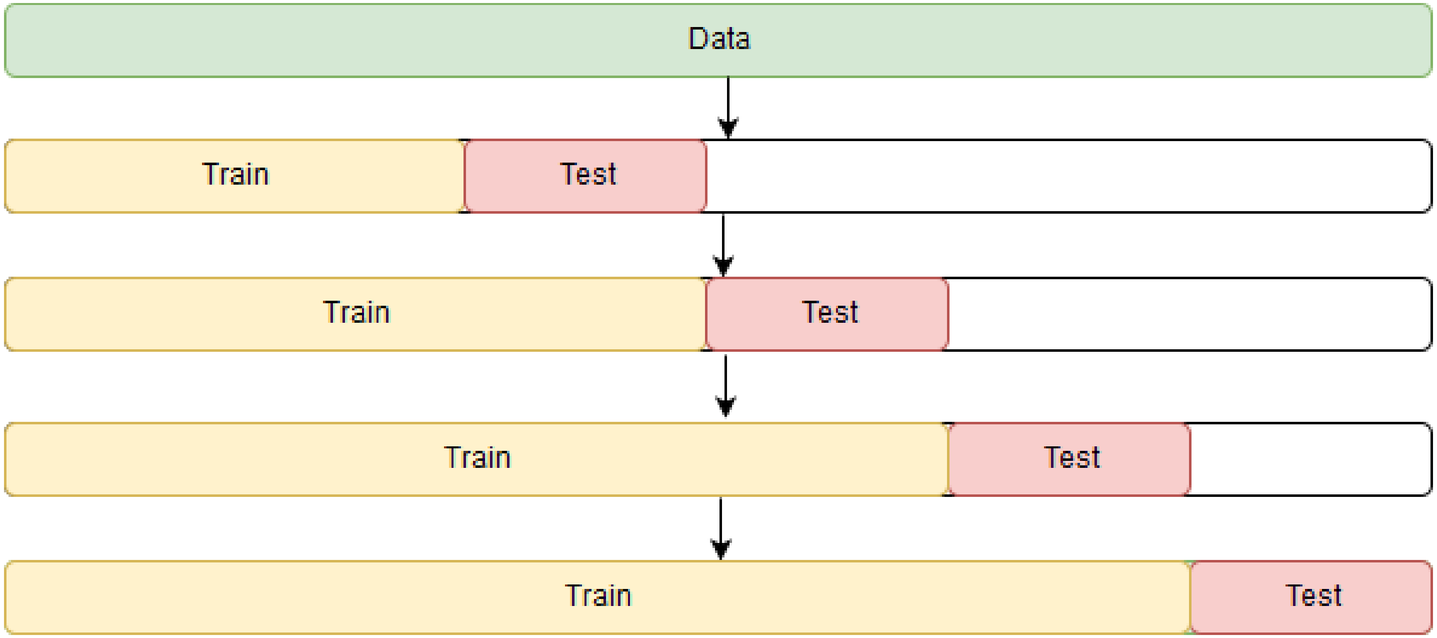


K-Fold Strategy



Data Splitting for Time Series

- Time series data requires careful splitting due to the presence of correlations in data
- Split by time



Jupyter Notebook

- *Case Study*

ON-BRAND STATEMENT

FOR GENERAL USE

> What defines the students and faculty of the University of Washington? Above all, it's our belief in possibility and our unshakable optimism. It's a connection to others, both near and far. It's a hunger that pushes us to tackle challenges and pursue progress. It's the conviction that together we can create a world of good. And it's our determination to Be Boundless. Join the journey at **uw.edu**.