

Data Science

# TERMINAL LEARNING OBJECTIVE

Students will understand the fundamentals of data science

Given classroom environment, instructor assistance, workstation, physical and virtual ranges, and access to the training materials, students will be able to do what is described in the terminal learning objective.

# Agenda

Friday 23JUL21

- LSA1: Understand data science.
- LSA2: Understand the components of Data Science statistics.
- LSA3: Understand Data Processing in Data Science.
- LSA5: Understand Data Science Metrics and Techniques.
- LSA6: Understand the concepts associated with Data Analysis.
- LSA7: Understand the concepts associated with Data Visualization.

## LSA1: Understanding Data Science

**Outcome:** At the end of this learning step, students will be able to define data science.

What is data science?

Data Science is the study of the  
generalizable extraction of  
knowledge from data.<sup>[1]</sup>

# DATA SCIENCE LANDSCAPE



BY: CHANIN NANTASENAMAT

DATA PROFESSOR

<http://youtube.com/dataprofessor>

FEBRUARY 14, 2020

# THE DATA SCIENCE PROCESS



Data Engineers

Data Analysts

Machine Learning Engineers

Data Scientists

(Drawn by Chanin Nantasenamat in collaboration with Ken Jee)



# Data Science Applications

- Voice Recognition
  - Siri, Google Voice, Alexa
- Fraud detection
- Medical Image Analysis
- Sports
  - Find undervalued players (Think Money Ball) Liverpool did this and won the 2019 Premier League championship
  - Determine the best positions on the field or court
- E-Commerce
  - Automated Ad placement
  - Product Recommendation
- Social Media
  - LinkedIn recommend connections
  - Tinder recommends Matches
  - Facebook presents potential connections

# Voice Recognition



# Video recommendation systems



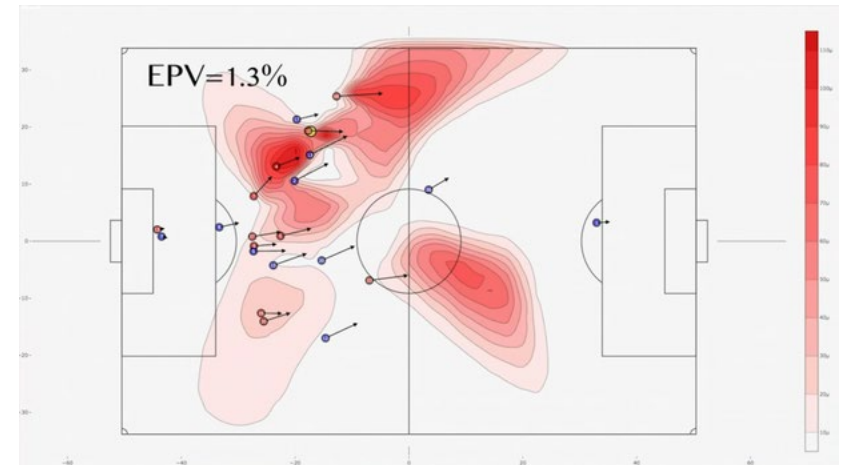
# E-Commerce Applications

- Product Recommendations
- Personalized Advertisement

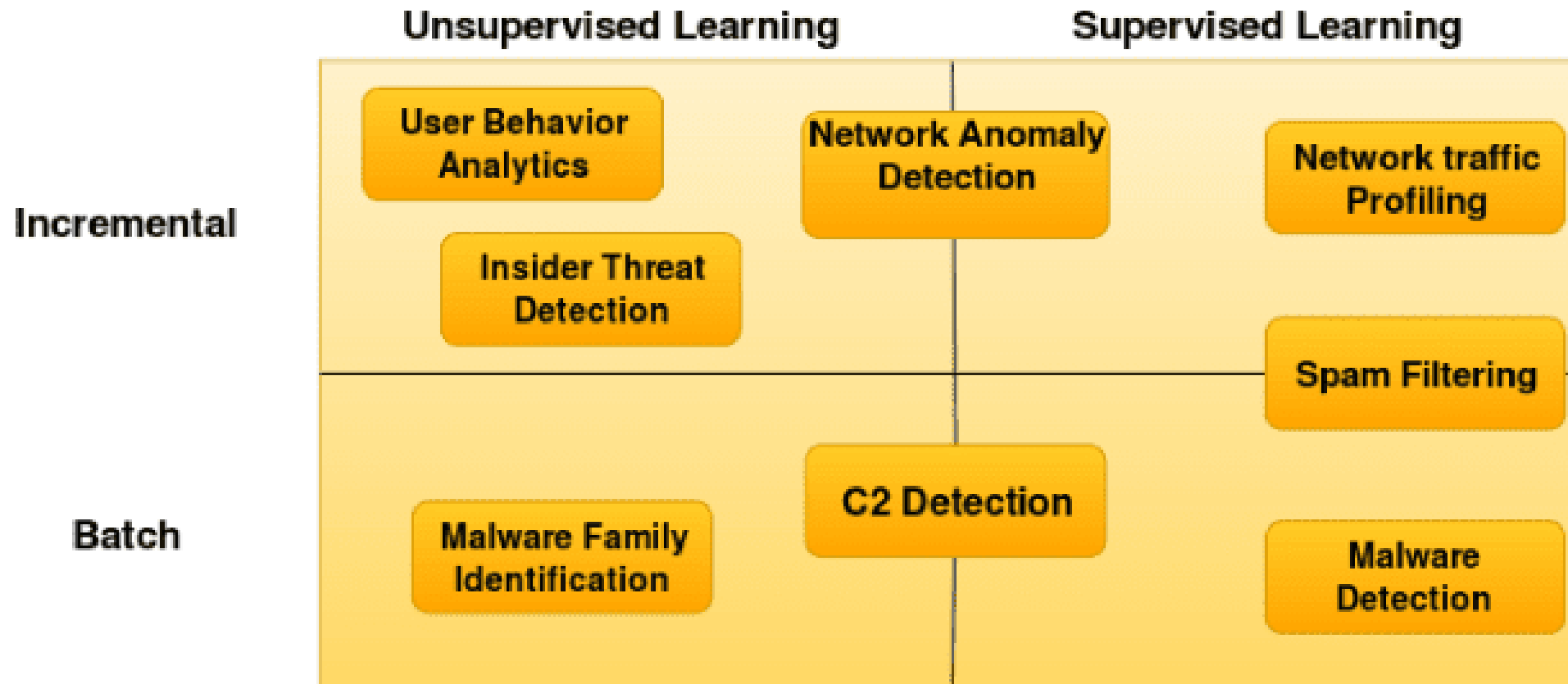


# Sports – Liverpool 2019

- Find undervalued players (2019 Liverpool)
- Determine best strategy on the field that increases goal probability
- Won 2019-2020 Premier League title



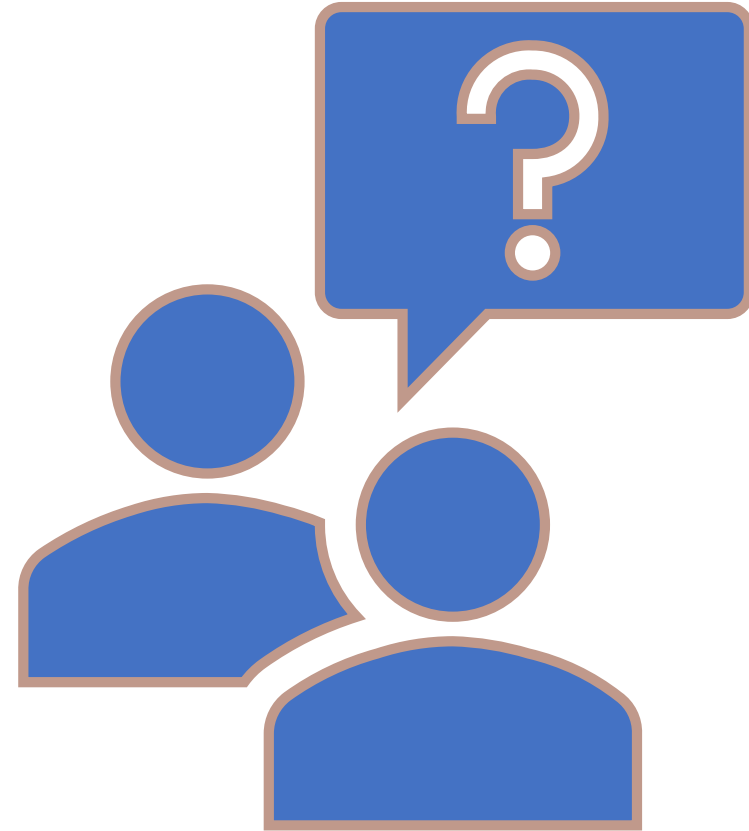
# Cyber Security Applications



# Conclusions

- Data Science spans multiple disciplines and seeks to find actionable insights from data
- Applications exist in every domain

# Questions





**LSA2:** Understand the components of Data Science statistics.

**Outcome:** At the end of this learning step, students will be able to identify and define the components of data science statistics.

# Central Tendency

- Used to get a simple understanding of a feature or variable
- Informs how biased a dataset is.
- Utilize Mean and/or Median to determine central tendency
- Median is more robust to outliers

# Mean – average value

$$\text{Mean} = \frac{\text{Sum of all data values}}{\text{Number of data values}}$$

$$\bar{x} = \frac{\sum x}{n}$$

$$\bar{x} = \frac{1 + 2 + 3 + 4 + 5}{5} = \frac{15}{5} = 3$$

```
import statistics as stats  
array = [1, 2, 3, 4, 5]  
stats.mean(example_array)
```

# Median – middle value

Odd list length:

- Median is  $\frac{n+1}{2}$  element in ordered list.

Even list length:

- Median is the average of the  $\frac{n}{2}$  and  $\frac{n+1}{2}$  elements in ordered list.

```
import statistics as  
stats  
  
array = [1, 2, 3, 3, 4,  
5]  
  
stats.median(example_ar  
ray)  
  
median = 3
```

# MODE

The most frequent element.

```
import statistics as  
stats  
  
array = [1, 2, 3, 3, 4,  
5]  
  
stats.mode(example_array)  
  
mode = 3
```

# Quartiles

- Three values that split the provided data into four equal parts (25%, 50%, 75%).
- Gives us insight to if a value is an outlier or not (Typically in the top/bottom 5%).

```
import statistics as  
stats  
  
array = [1, 2, 3, 4, 9,  
5]  
  
stats.quantiles(array,  
n=4) =  
[1.75, 3.5, 6.0]
```

# Variance

- Measurement to determine how dispersed individual elements are from the mean.
- Captures variability of elements from the mean

$$\sigma^2 = \frac{\sum_{i=1}^n X_i - \mu}{n}$$

- $\sigma^2$  = Variance
- $X_i$  = Element  $i$
- $\mu$  = population mean

```
import statistics as  
stats  
  
array = [1, 2, 3, 4, 9,  
5]  
  
stats.variance(array) =  
8
```

# Standard Deviation

- Measurement to determine how dispersed individual elements are relative to the mean.
- Square root of the Variance

$$\sigma = \sqrt{\frac{\sum_{i=1}^n X_i - \mu}{n}}$$

- $\sigma$  = Standard Deviation
- $X_i$  = Element  $i$
- $\mu$  = population mean
- $n$  = number of elements

```
import statistics as stats  
  
array = [1, 2, 3, 4, 9, 5]  
  
stats.stdev(array) =  
2.82843
```



# Covariance

- Measures the relationship between two variables.
- Does not measure dependency between two variables.
- Positive Covariance
  - Two variables go in same direction
- Negative Covariance
  - Two variables go in opposite directions

$$COV(X, Y) = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{n - 1}$$

```
from numpy import cov
```

```
array1 = [1, 2, 3, 4, 9, 5]
```

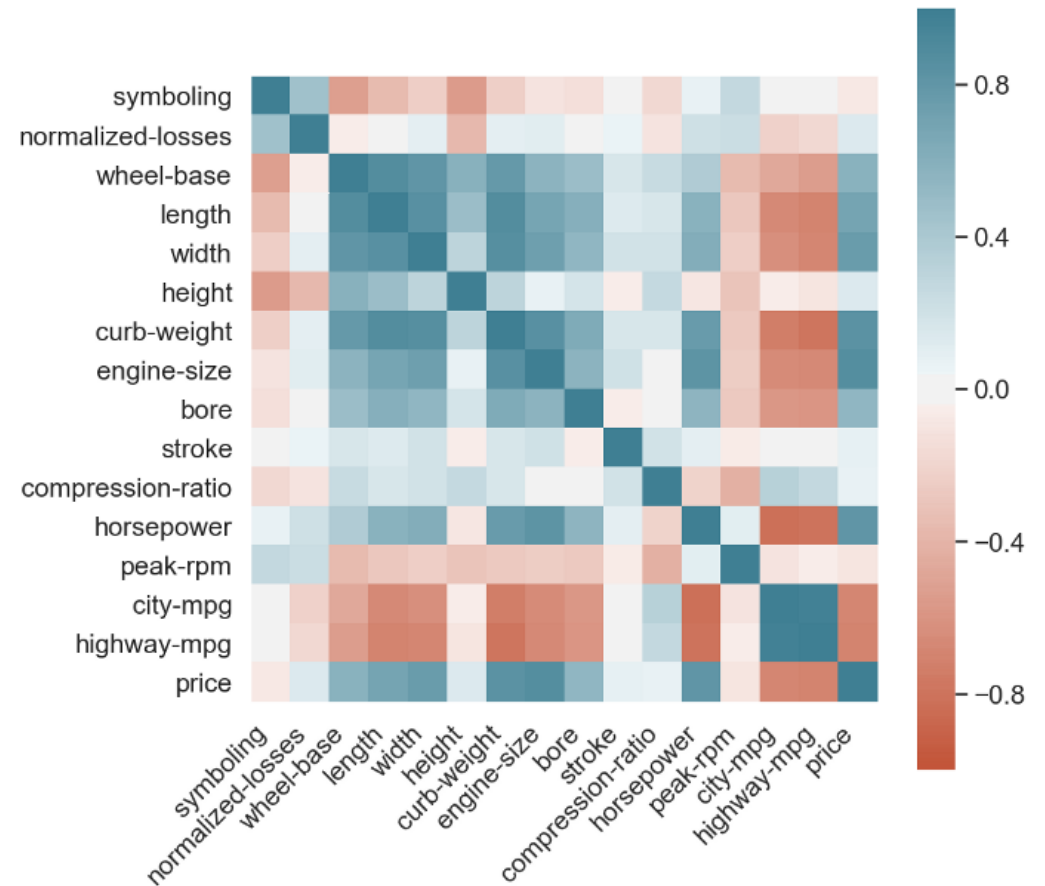
```
array2 = [1, 2, 3, 4, 9, 5]
```

```
cov(array1, array2) = 1
```

- $COV(X, Y)$  = covariance of  $X$  and  $Y$
- $X_i$  = the values of  $X$
- $\bar{X}$  = the sample mean of  $X$
- $Y_i$  = the values of  $Y$
- $\bar{Y}$  = the sample mean of  $Y$
- $n$  = number of elements

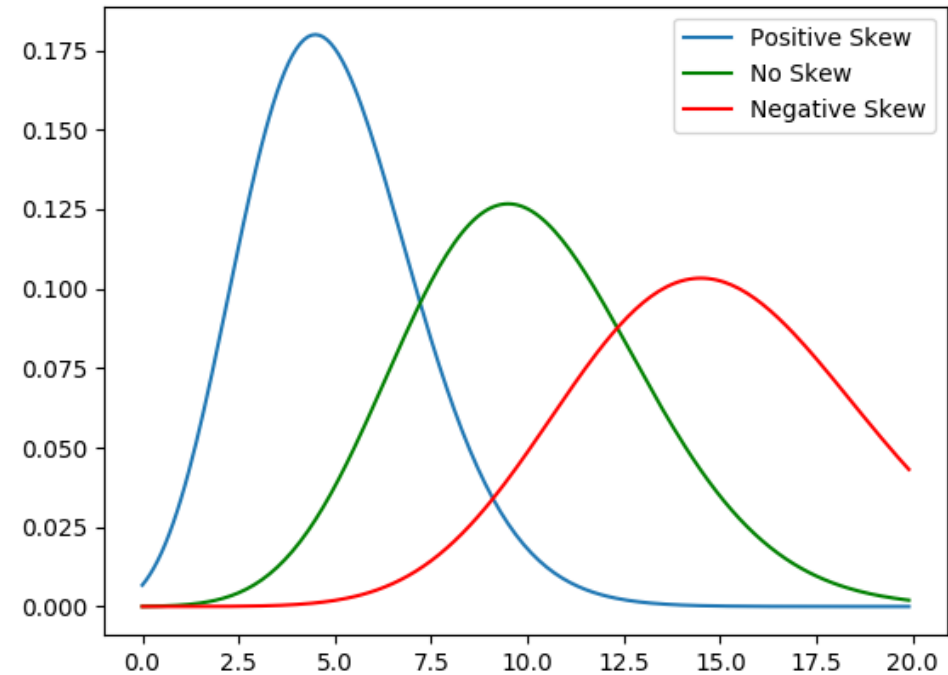
# Correlation

- Normalized covariance (-1.0, 1.0)
- If the correlation between two variable is 1, then the change in variable one results in a proportional move in the same direction for variable 2.
- Useful for Principal Component Analysis (PCA) or other dimensionality reduction techniques.



# Skewness

- Measures asymmetry
  - Positive value means skewed to the left.
  - Negative value means skewed to the right
- $$skewness = \frac{3 \times (Mean - Median)}{Standard\ Deviation}$$
- Provides insight to if the distribution is Gaussian
  - `scipy.stats.skew(array)`



**LSA3:** Understand Data Processing in Data Science.

**Outcome:** At the end of this learning step, students will be able to identify and define data processing techniques.

# What is Data Preprocessing/processing?

A data science procedure in which data transforms into a clean, machine-interpretable format.

# Encoding schemes

- One hot encoding: Convert label encoding to a non-ordering relationship variable
- Binarization: Convert variable into 0s and 1s based on a fixed threshold.
- Discretization: Split a continuous variable into categories or groups
  - Equal Width, Equal Frequency, K means

# Normalization / Rescale

Normalization gives equal weight/importance to each variable

- (Rescaling) Min-Max normalization
- Mean normalization
- Z-score normalization

# Processing values

## Missing Values

- Remove entries with missing values
- Impute values for those missing

## Duplicate Values

- Remove Duplicates

## Inconsistent Values

- Create consistent format (Date, Address, Phone number, etc.)
- Outliers
  - Remove



# Why do we need to Process Data?

- Missing, Duplicate, Inconsistent values
- Outliers
- Feature Enrichment
- Feature Encoding
- Dimensionality Reduction
- Imbalanced data

# One Hot Encoding

Car Maker	Categorical Value	Price
Passat	1	12000
Civic	2	10000
Accord	3	14000
Accord	3	15000

**Label Encoding**



Passat	Civic	Accord	Price
1	0	0	1200
0	1	0	10000
0	0	1	14000
0	0	1	15000

**One-hot Encoding**

# Binarization

Passat	Civic	Accord	Price
1	0	0	1200
0	1	0	1000
0	0	1	14000
0	0	1	15000



Passat	Civic	Accord	Price
1	0	0	0
0	1	0	0
0	0	1	1
0	0	1	1

# Discretization

- Transforming continuous data into discrete data (categories or bins)
- Height is a continuous variable that can be transformed into short, average, tall bins
  - Short (less than 60 in)
  - Average ( between 60-71 inches)
  - Tall (greater than 71 inches)

# Why Transformations?

- Utilized to make your data work better for you
- Most raw data is skewed, but some models ( linear and logistic regression) follow normal distributions.
- Map a skewed distribution to a normal distribution through a transform

# Log Transform

- Attempt to get distribution close to a normal distribution
- Not applied to features with negative values
- Applied for skewed-right data.
- Scale changes to multiplicative scale (linearly distributed data)

# Square Root Transform

- Attempt to get distribution close to a normal distribution
- Compresses high values, so lower values are more spread out.
- Could apply this to the dependent variable, but results become less interpretable.
- Useful for right skewed distributions.

# Square Transform

- Attempt to get distribution close to a normal distribution
- Useful for left skewed distributions.



# Feature enrichment

The addition and/or transformation of features to increase the variance within the dataset to produce more robust models.

- Supplement data with data from other sources
- Feature Selection
- Feature Transformation
- Data Imputation

# Questions



## LSA5: Understand Data Science Metrics and Techniques.

**Outcome:** At the end of this learning step, students will be able to identify, define, and apply data science metrics and techniques.

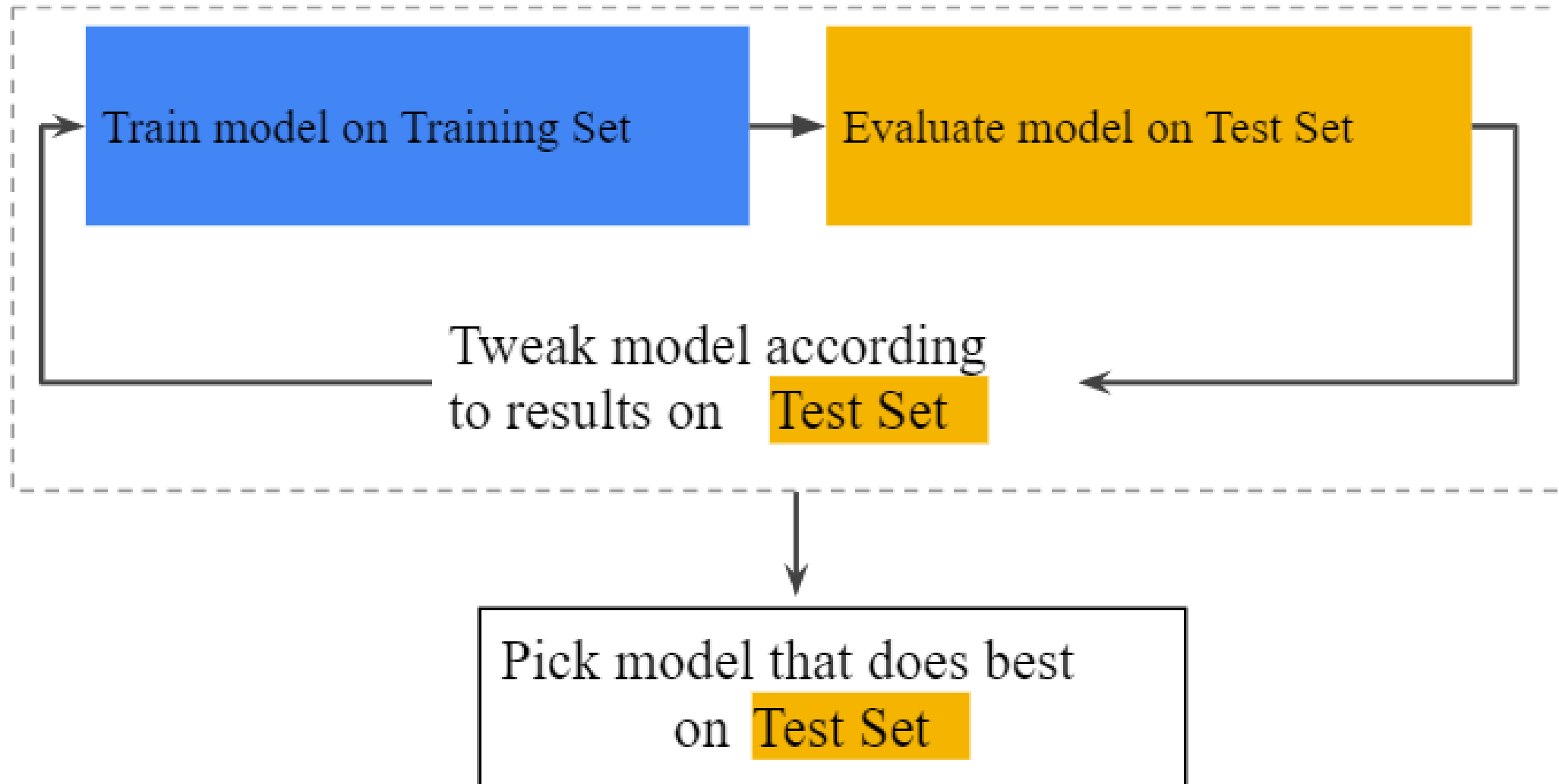
# Agenda

- Models
- Model Evaluation Techniques
- Performance Metrics

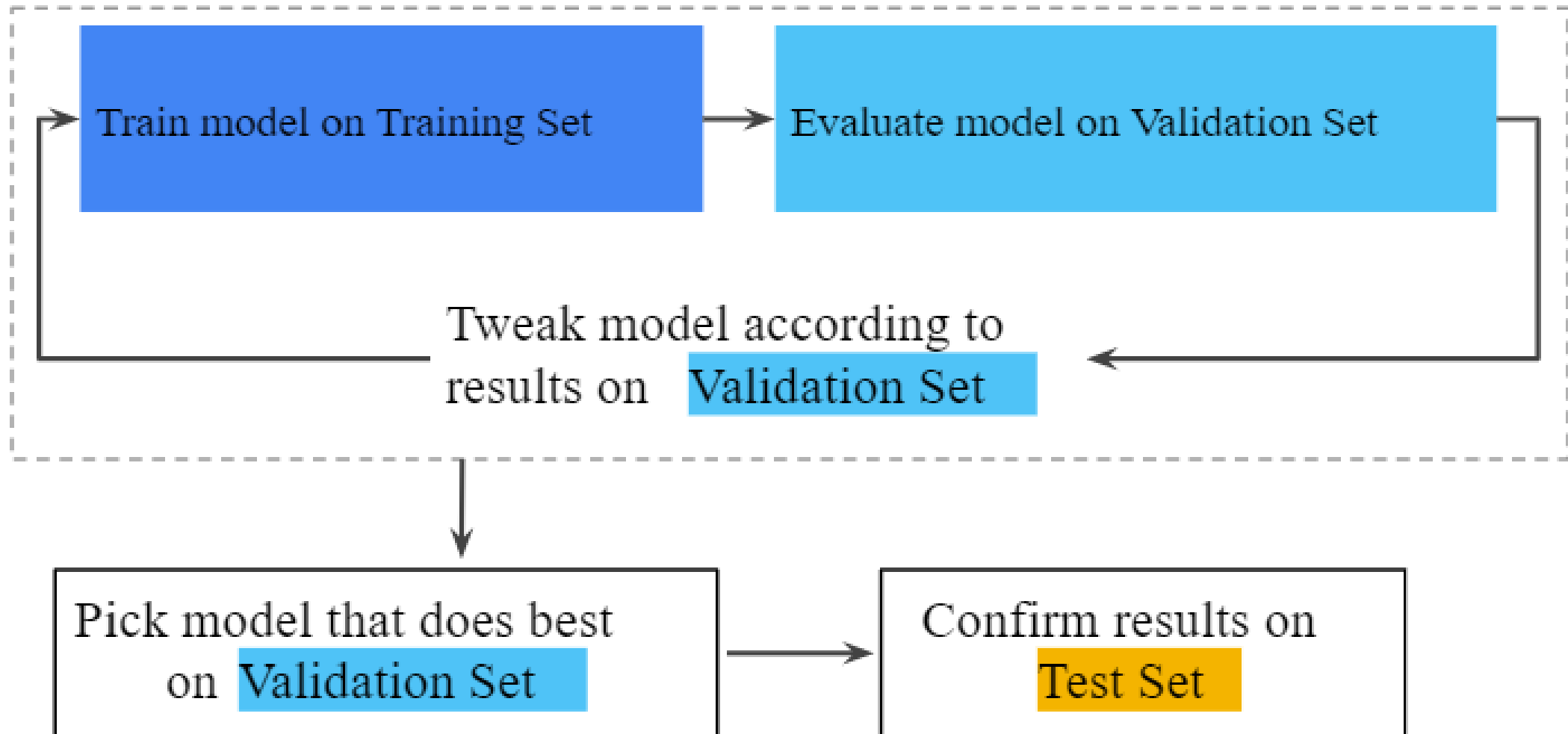
# Models

- Define relationships between features and label (continuous or discrete)
- Training: provide labeled examples and learn the relationship between features and label
- Inference: apply the trained model to predict the label for new unlabeled examples

# Model Evaluation Technique: Train, Test



# Model Evaluation Technique: Train, Validation, Test



# Model Evaluation Technique: Cross-Validation

- Split data into  $k$ -splits.
- Hold out one split as the test data and use the remaining  $k-1$  splits as test data.
- Record the performance metric for model
- Repeat  $k$  times
- Less variance in performance across  $k$  models results in higher validation of model



# Model Evaluation Technique: Parameter Tuning

- There are several parameters that can hold different values
- We must decide which values are best for our model.
- One method is grid search:
  - Take all combinations of desired parameters and test each one. Select the parameter combination that yields the highest performance value for the selected performance metric.

# Performance metrics for classification models

# Performance Metrics: Definitions

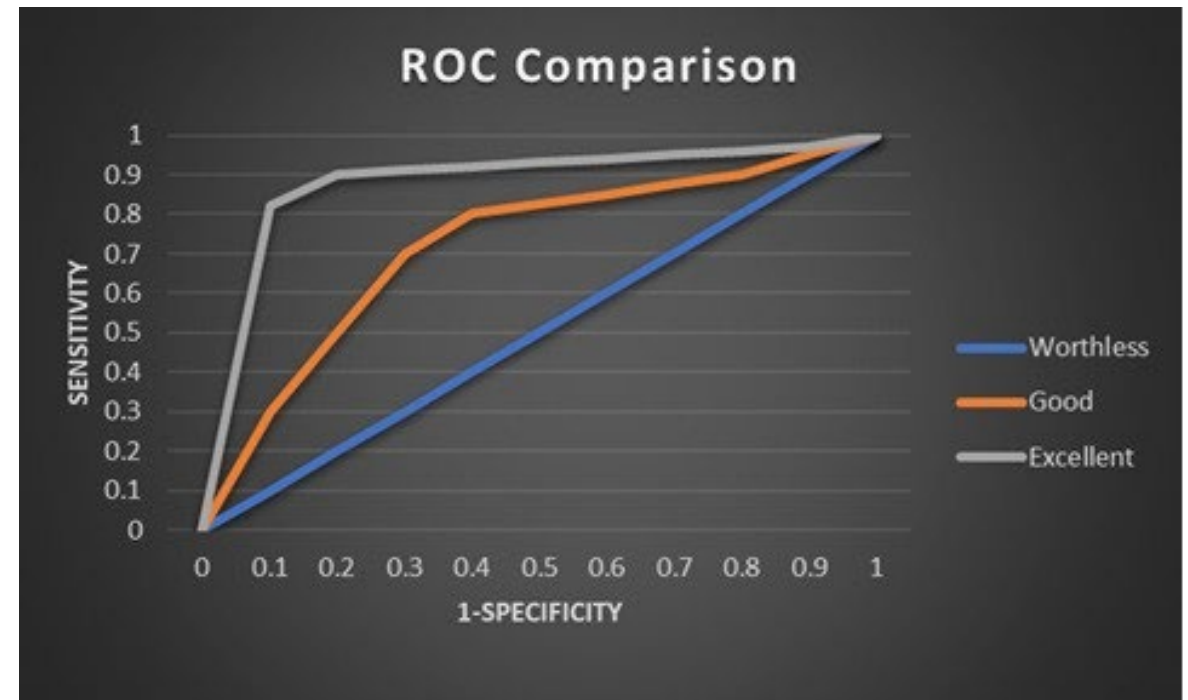
- **True Positive:** outcome where the model correctly predicts the positive class.
- **True Negative:** outcome where the model correctly predicts the negative class.
- **False Positive:** outcome where the model incorrectly predicts the positive class.
- **False Negative:** outcome where the model incorrectly predicts the negative class.

# Performance Metrics: Thresholding

- Decision threshold: a value that determines whether an entry is positive or negative
- Usual assumption is that the threshold should always be 0.5
- Not always the case
- Did a student pass the class or not? ( Threshold at 0.7?)

# Performance Metric: ROC Curve and AUC

- ROC Curve shows performance at all classification thresholds
- AUC is the area underneath the ROC Curve
- AUC is the probability that a random positive example is ranked higher than a random negative sample
- AUC is scale-invariant (ranking not absolute values)



# Performance Metrics: Accuracy

- Evaluation metric for classification models
- Fraction of predictions the model got right
- $Accuracy = \frac{\text{Number of right predicts}}{\text{Total umber of predictions}}$
- Watch out! When working with class-imbalanced data sets

# Performance Metric: Accuracy

- Model to predict cancer in 100 patients
- 91 Benign, 9 Cancer
- Accuracy is 91%
- Is this a good model?
- Only 1/9 cancer patients were identified.
- “Our model predicted 91% of patients correctly.”

## True Positive (TP):

- Truth: Cancer
- Prediction: Cancer
- TP: 1

## False Positive (FP):

- Truth: Benign
- Prediction: Cancer
- TP: 1

## False Negative (FN):

- Truth: Cancer
- Prediction: Benign
- TP: 8

## True Negative (TN):

- Truth: Benign
- Prediction: Benign
- TP: 90

# Performance Metric: Precision

- What proportion of positive predictions was correct?
- $\text{Precision} = \frac{TP}{TP+FP}$
- $\text{Precision} = \frac{1}{1+1} = 0.5$
- “When the model predicts a patient has cancer, the model is correct 50% of the time.”

<b>True Positive (TP):</b> <ul style="list-style-type: none"><li>• Truth: Cancer</li><li>• Prediction: Cancer</li><li>• TP: 1</li></ul>	<b>False Positive (FP):</b> <ul style="list-style-type: none"><li>• Truth: Benign</li><li>• Prediction: Cancer</li><li>• TP: 1</li></ul>
<b>False Negative (FN):</b> <ul style="list-style-type: none"><li>• Truth: Cancer</li><li>• Prediction: Benign</li><li>• TP: 8</li></ul>	<b>True Negative (TN):</b> <ul style="list-style-type: none"><li>• Truth: Benign</li><li>• Prediction: Benign</li><li>• TP: 90</li></ul>



# Performance Metric: Recall

- What proportion of actual positives were identified correctly?
- $\text{Recall} = \frac{TP}{TP+FN}$
- $\text{Recall} = \frac{1}{1+8} = 0.11$
- “Our model identifies 11% of cancer patients.”

<b>True Positive (TP):</b> <ul style="list-style-type: none"><li>• Truth: Cancer</li><li>• Prediction: Cancer</li><li>• TP: 1</li></ul>	<b>False Positive (FP):</b> <ul style="list-style-type: none"><li>• Truth: Benign</li><li>• Prediction: Cancer</li><li>• TP: 1</li></ul>
<b>False Negative (FN):</b> <ul style="list-style-type: none"><li>• Truth: Cancer</li><li>• Prediction: Benign</li><li>• TP: 8</li></ul>	<b>True Negative (TN):</b> <ul style="list-style-type: none"><li>• Truth: Benign</li><li>• Prediction: Benign</li><li>• TP: 90</li></ul>

# Performance Metric: F1-Score

- The harmonic mean of precision and recall
- $F_1 = 2 \frac{precision * recall}{precision + recall}$
- $F_1 = \frac{0.11 * 0.5}{0.11 + 0.5} = 0.09$
- “Our model has an F1-score of 0.09.”

<b>True Positive (TP):</b> <ul style="list-style-type: none"><li>• Truth: Cancer</li><li>• Prediction: Cancer</li><li>• TP: 1</li></ul>	<b>False Positive (FP):</b> <ul style="list-style-type: none"><li>• Truth: Benign</li><li>• Prediction: Cancer</li><li>• TP: 1</li></ul>
<b>False Negative (FN):</b> <ul style="list-style-type: none"><li>• Truth: Cancer</li><li>• Prediction: Benign</li><li>• TP: 8</li></ul>	<b>True Negative (TN):</b> <ul style="list-style-type: none"><li>• Truth: Benign</li><li>• Prediction: Benign</li><li>• TP: 90</li></ul>

# Performance metrics for regression models

# Performance Metric: Mean Squared Error

- Mean squared difference between the target and predicted value
- Preferred because it is differentiable and can be optimized better
- Penalizes even small errors

$$MSE = \frac{1}{n} \sum \left( \underbrace{y - \hat{y}}_{\substack{\text{The square of the difference} \\ \text{between actual and} \\ \text{predicted}}} \right)^2$$

# Performance Metric: Root Mean Squared Error

- Square root of MSE
- High penalty for large errors

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (Predicted_i - Actual_i)^2}{N}}$$

# Performance Metric: Mean Absolute Error

- Absolute difference between the target and the predicted value
- More robust to outliers
- Penalty is not as strong as MSE

The diagram illustrates the Mean Absolute Error (MAE) formula with the following components and annotations:

- Divide by the total number of data points:** A blue line points to the fraction  $\frac{1}{n}$ , which is enclosed in a blue box.
- Sum of:** A blue line points to the summation symbol  $\Sigma$ .
- Actual output value:** A green line points to the variable  $y$ , which is enclosed in a green box.
- Predicted output value:** An orange line points to the variable  $\hat{y}$ , which is enclosed in an orange box.
- The absolute value of the residual:** A bracket underneath the subtraction  $y - \hat{y}$  points to the absolute value bars  $| \cdot |$ .

$$MAE = \frac{1}{n} \sum |y - \hat{y}|$$

# Questions



**LSA6:** Understand the concepts associated with  
Data Analysis.

**Outcome:** At the end of this learning step,  
students will understand data analysis concepts.



# Exploratory Data Analysis (EDA)

- Investigate data sets and summarize their main characteristics
- Utilize statistics and data visualization methods
- Guides how to manipulate the data to best answer the problem you are addressing
- Detect outliers or anomalous events before modeling

# Why do EDA?

- Inform assumptions
- Identify obvious patterns, outliers, and variable relationships
- Confirms you have the right data you need for the problem
- Identify models not correct for the problem

# Four types of EDA

- Univariate non-graphical
- Univariate graphical
- Multivariate nongraphical
- Multivariate graphical

# Univariate non-graphical EDA

- Examine one variable
- Does not look at correlation or relationships
- Describes the data using mean, median, min, max, quartiles, standard deviation.

# Univariate Graphical EDA

- Examine one variable
- Does not look at correlation or relationships
- Histogram
- Box plot

# Multivariate non-graphical EDA

- Examine multiple variables
- Looks at correlation or relationships
- Typically, correlation and covariance.

# Multivariate Graphical EDA

- Examine multiple variables
- Utilizes graphics to display relationships between multiple variables or data sets
- Scatter Plot
- Multivariate bar plot or histogram
- Heat map

**LSA7:** Understand the concepts associated with Data Visualization.

**Outcome:** At the end of this learning step, students will identify data visualization concepts and create data visualizations.



# What are the common elements of a graphic?

- Some type of data
- Maps data to aspects of the plot
- Geometric in nature
- Statistical transformation
- Coordinate systems

# How are these plots Related?

- Scatter Plot

- Maps variables to x-axis and y-axis
- Uses points to represent each observation

- Line Plot

- Maps variables to x-axis and y-axis
- Uses lines to connect each observation

- Histogram

- Maps bins to x-axis and frequencies to y-axis
- Uses bars to represent bin frequency

- Bar Plot

- Maps categorical variable to x-axis and counts to y-axis
- Uses bars to represent category frequency

- Box Plot

- Maps 5-number summary ( min, lower-hinge, median, upper-hinge, max) to y-axis
- Uses shapes to represent values (box and whiskers)

# How are these plots Related?

- Contour Plot

- Maps three-dimensional surface to a two-dimensional surface
- Maps variables to x-axis and y-axis
- Uses contour lines to connect points of same response value
- Color bands between contours to represent range of values

- Facets

- Matrix of two-dimensional graphs partitioned by a third variable

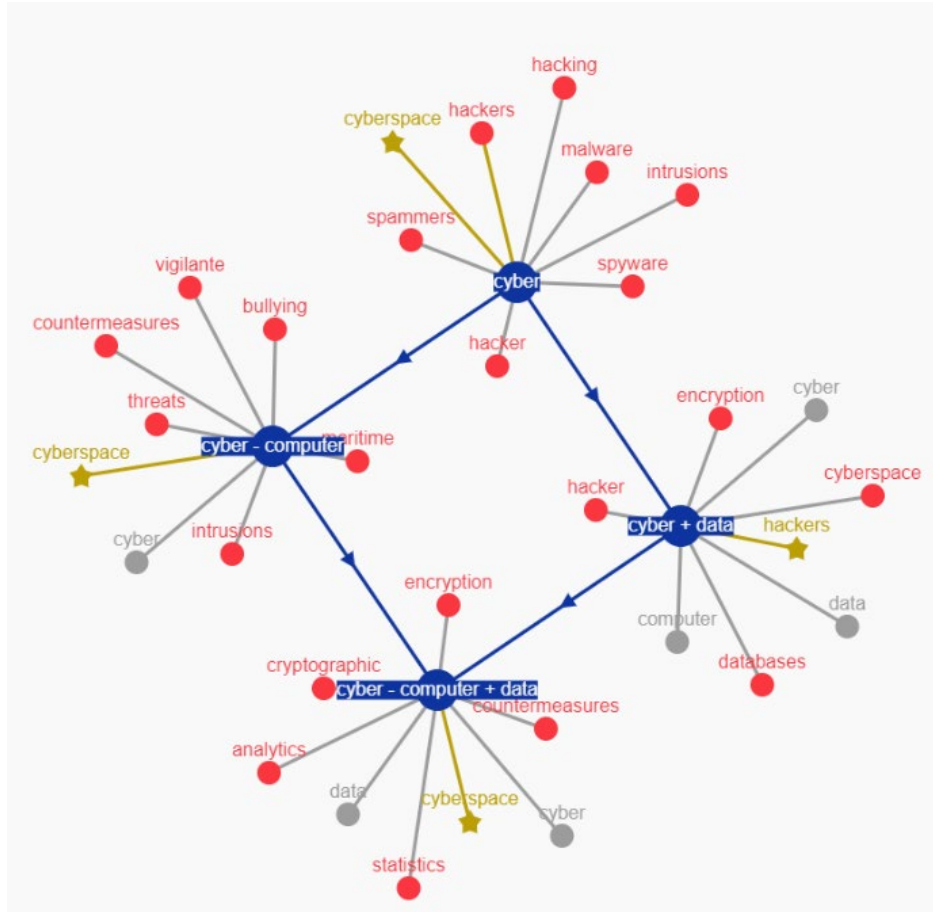
# Python packages for Visualization

- Matplotlib
- Seaborn
- Plotly
- Dash

# Application for Data Visualization

- Exploratory Data Analysis
- Present results
- Interactive user application

# Application for Data Visualization



- <https://dash-gallery.plotly.host/dash-word-arithmetic/>

**LSA4:** Demonstrate the ability to solve statistical and data processing problems.

**Outcome:** At the end of this learning step, students will solve statistical and data processing problems.

**LSA8:** Demonstrate the ability use metrics, data analysis and visualization techniques to understand data

**Outcome:** At the end of this learning step, students will use metrics and visualization to analyze data.