Analytics & Machine Learning for Pipeline Integrity & Risk

Michael P. Gloven, P.E.
Pipeline-Risk (PLR)
Engineering Solutions & Software
www.pipeline-risk.com

Analytics & Machine Learning for Pipeline Integrity & Risk

Introduction

PLR Introduction

Experience

- **30+** yrs experience in integrity and risk management
- **200,000+** miles of transmission and distribution pipe analyzed
- **20+** integrity use cases
- **7 of top 20** midstream operators use our services
- Multiple industry **presentations** and **publications**

Technology

- Machine learning analytics **platform** purposed for pipeline systems
- Azure cloud-based **secure** infrastructure
- **Proven** open-source machine learning and statistical packages
- Curated and **ready to use**

Solutions

- **Automated** model learning process for any pipeline use case
- Library of learned **reference** models for project primer
- Models tuned to client **objectives** and data
- Onsite and open-enrollment **training classes**
- Regulatory **audit** acceptance

Course Objectives

- Learn machine learning fundamentals through inte
integrity & r
- Gain a pract
machine lea
- Discuss prac
the practice
- Provide a st
educational
learning

PLR Model Library

Group	Name	Theme	Training_Source	Assets
	All	All	All	All
	Index Threat - Equipment	Index	PLR Expertise & 100,000 miles+ of Projects	Any_Pipeline
	CO2 Dispersion Model	Industry Reference	Canary Analysis	Hazardous_Liquids
	Index Threat - Internal Corrosion	Index	PLR Expertise & 100,000 miles+ of Projects	Any_Pipeline
	Index Threat - Construction	Index	PLR Expertise & 100,000 miles+ of Projects	Any_Pipeline
	Index Consequence - Population	Index	PLR Expertise & 100,000 miles+ of Projects	Gas_Transmission
	Predict External Corrosion CGR	Threats	PLR Expertise & Large Projects based on MFL Data	Any_Pipeline
	PIR CFER Model	Industry Reference	CFER	Gas_Transmission
	Predict Incorrect Operations Susceptibility	Threats	PLR Expertise	Facilities Any_Pipeline
	Index Threat - Manufacturing	Index	PLR Expertise & 100,000 miles+ of Projects	Any_Pipeline
	Predict Missing Coating Types	Data Quality	PLR Expertise in Support of Training Course Multi-Class	Any_Pipeline
	Index Consequence - Business	Index	PLR Expertise & 100,000 miles+ of Projects	Gas_Transmission
	Index Consequence - Environment	Index	PLR Expertise & 100,000 miles+ of Projects	Gas_Transmission

“About You & Your Requirements”

Agenda

DAY 1

Section	Topic	Schedule
BASICS	Introduction 1.1 - Overall Process 1.2 - Common Questions 1.3 - Learning Categories 1.4 - Learning Methods 1.5 – The Math (opt)	8:00 – 9:45
	BREAK	9:45 – 10:00
	1.6 - Classification & Performance 1.7 - Classification Example	10:00 – 11:30
	LUNCH	11:30 – 12:30
	1.8 - Regression & Performance 1.9 - Regression Example 1.10 – Cross-Validation (Resampling)	12:30 – 2:00
	BREAK	2:00 – 2:15
	1.11 - Model Explainability 1.12 - Explainability Example	2:15 – 2:45
DATA	2.1 – Training Data Introduction 2.2 – Data Integration Concepts 2.3 – Data Sampling 2.4 – Data Quality 2.5 – Data Pre-Processing	2:45 – 3:30
	BREAK	3:30 – 3:45
MODEL VALIDATION & TUNING	3.1 – Model Error 3.2 - Model Tuning 3.3 – Deterministic Model Validation Day 1 Closing & Questions	3:45 – 5:00

DAY 2

Section	Topic	schedule
USE CASES	Day 1 Recap & Questions Day 2 Introduction • Interactive use case discussion • Attendee selected use cases • Follow standard process Selected Use Cases	8:00 – 9:45
	BREAK	9:45 – 10:00
	• Selected Use Cases	10:00 – 11:30
	LUNCH	11:30 – 12:30
	• Software Instruction based on Use Case & Hands-On Requirements • Selected Use Cases (Optional Focus on Machine Learned based Risk)	12:30 – 2:00
	BREAK	2:00 – 2:15
	• Selected Use Cases (Optional Hands-On with ML.ai & Attendee or Example Data)	2:15 – 2:45
	BREAK	3:30 – 3:45
	• Selected Use Cases (Optional Hands-On with ML.ai & Attendee or Example Data)	3:45 – 4:30
	Course Closing & Questions	4:30 – 5:00

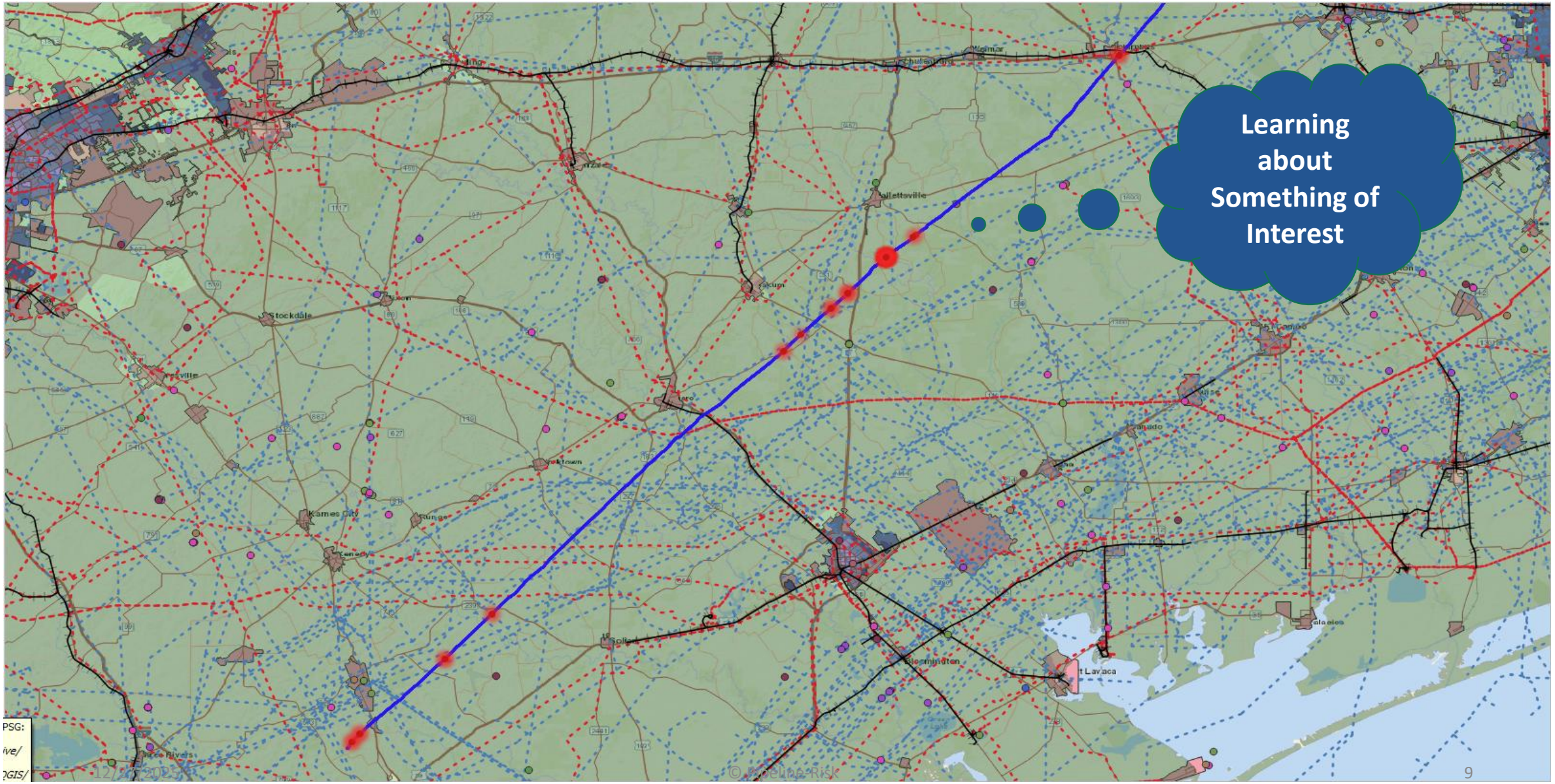
BASICS

Machine Learning Essentials

Unit 1.1

Overall Process

Typical Use Case



Training Data

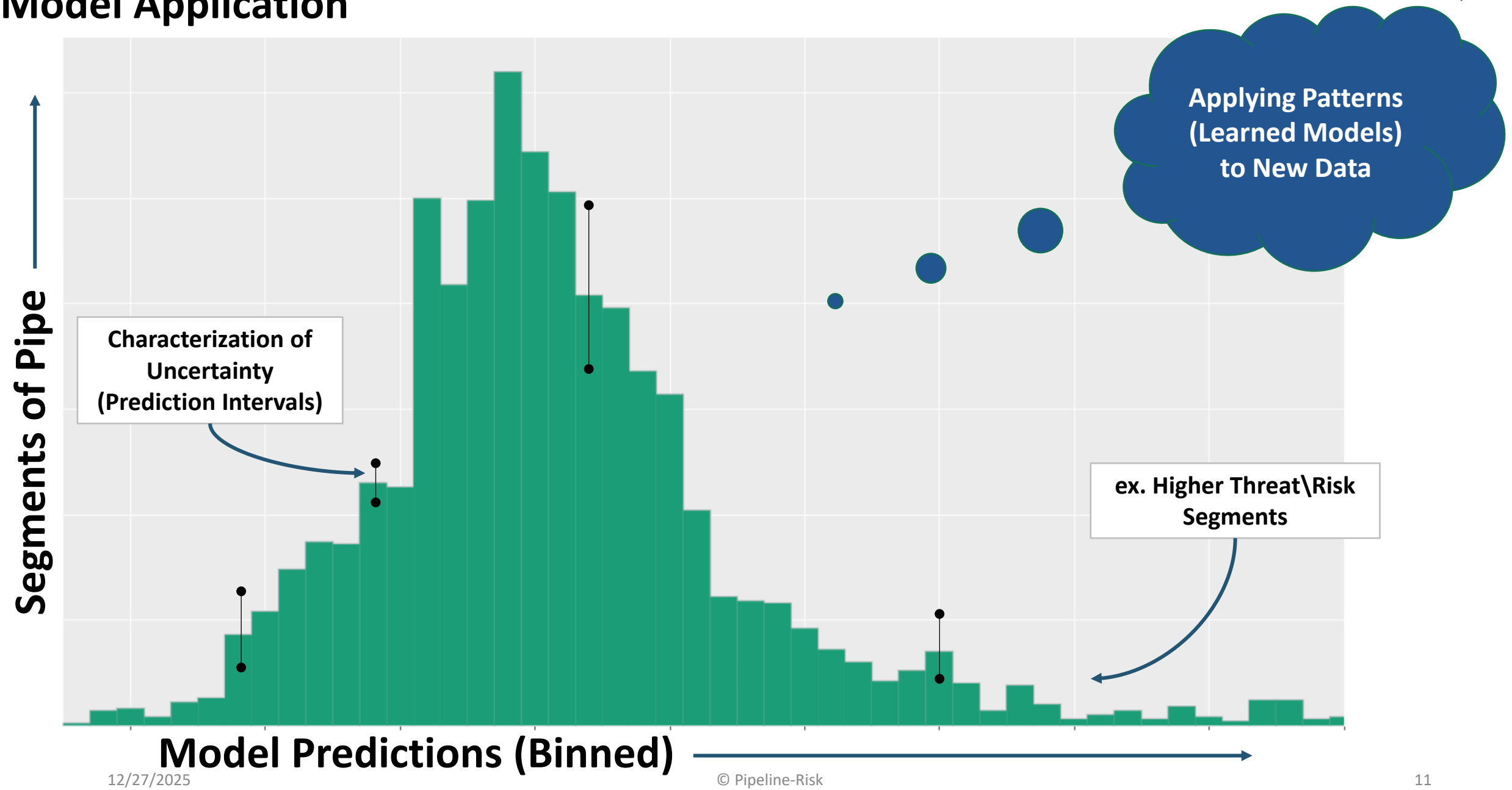
Learning Target

Predictor Data

Observation	TPD	Class	Diameter	DOC	Farmland	Install_Yr	LineMark	PatroFreq	RAILROAD	Structures
All	A	A	All	A	All	All	All	All	All	All
No_Evidence	F	1.00	8.00	24.00	Not_Farmland	1,980.00	Line_of_Site	Semi-Annual	None	
No_Evidence	F	1.00	8.00	24.00	Not_Farmland	1,980.00		Semi-Annual	No	
No_Evidence	F	1.00	8.00	25.00	Not_Farmland	1,980.00	Line_of_Site	Semi-Annual	No	
No_Evidence	F	1.00	8.00	33.00	Not_Farmland	1,980.00	Line_of_Site	Se nual	No	
No_Evidence	F	1.00	8.00	30.00	Not_Farmland	1,980.00	Lin f_Site	Semi-Annual	None	
One_Call_Violation	T	1.00	8.00	26.00	Not_Farmland	1,980.00	Line_of_Site	Semi-Annual	RR	Structures
One_Call_Violation	T	2.00	8.00	26.00	Not_Farmland	1,980.00	Line_of_Site	Semi-Annual	RR	Structures
Near_Miss	T	2.00	8.00	29.00	Farmland	1,980.00	Line_of_Site	Semi-Annual	RR	Structures
One_Call_Violation	T	2.00	8.00	24.00	Farmland	1,980.00	Line_of_Site	Semi-Annual	RR	Structures
Near_Miss	T	2.00	8.00	28.00	Farmland	1,980.00	Line_of_Site	Semi-Annual	None	None
Near_Miss	T	2.00	8.00	34.00	Farmland	1,980.00	Line_of_Site	Semi-Annual	None	None
Near_Miss	T	2.00	8.00	41.00	Farmland	1,980.00	Line_of_Site	Semi-Annual	None	None
No_Evidence	F	2.00	8.00	31.00	Farmland	1,980.00	Line_of_Site	Bi-Weekly	None	None
No_Evidence	F	3.00	8.00	24.00	Farmland	1,980.00	Line_of_Site	Bi-Weekly	None	None

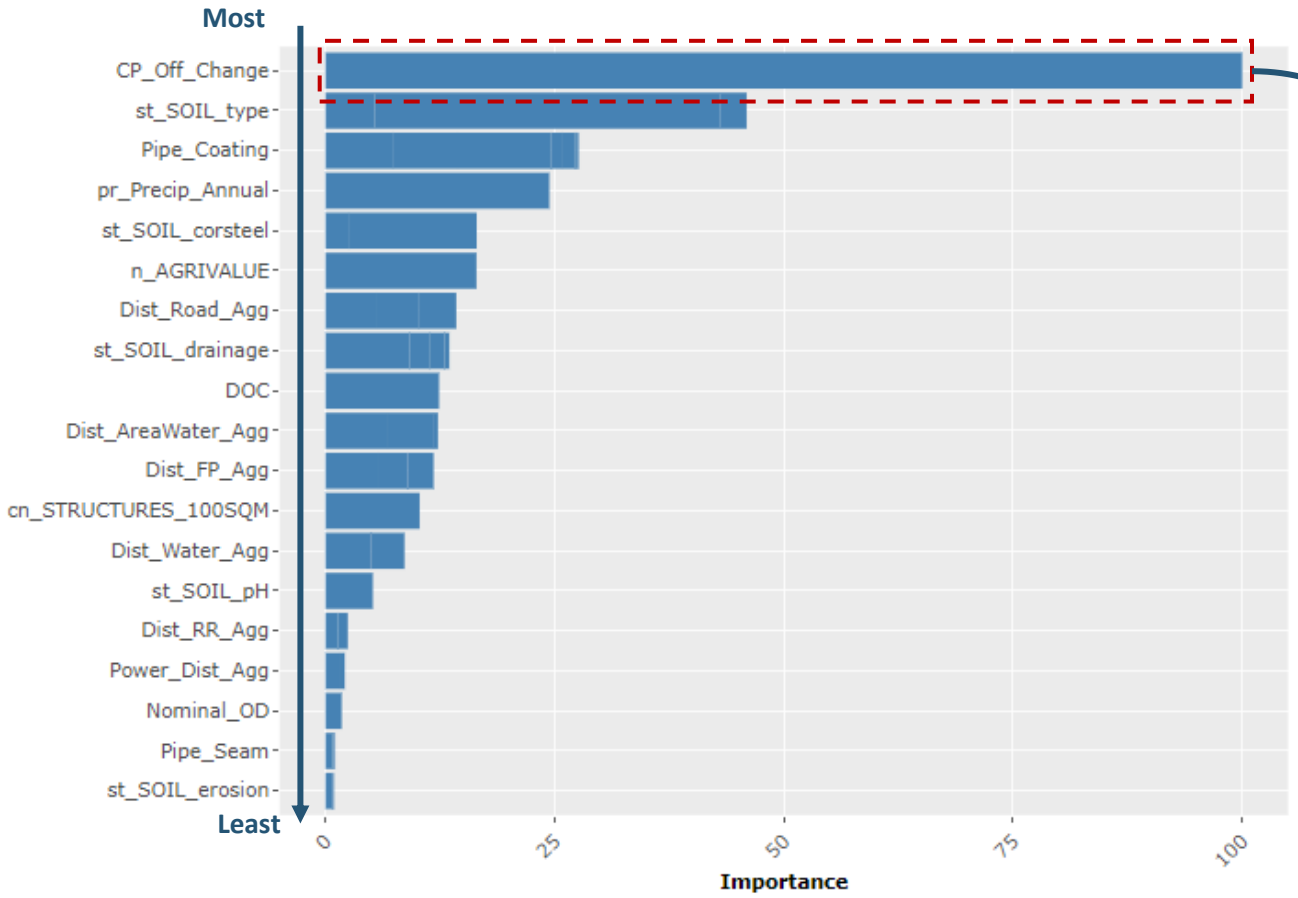
Finding Useful Patterns in Structured Data

Model Application

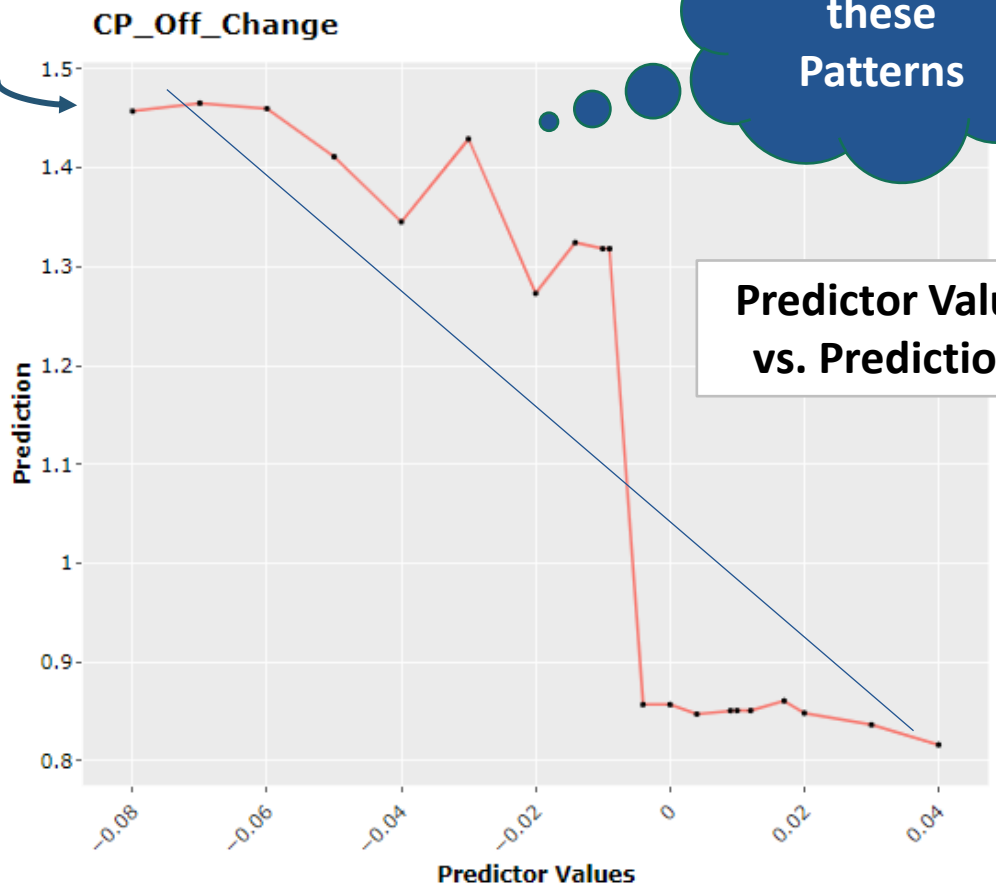


Explain Results

Predictor Importance



Predictor Directionality



Explaining these Patterns

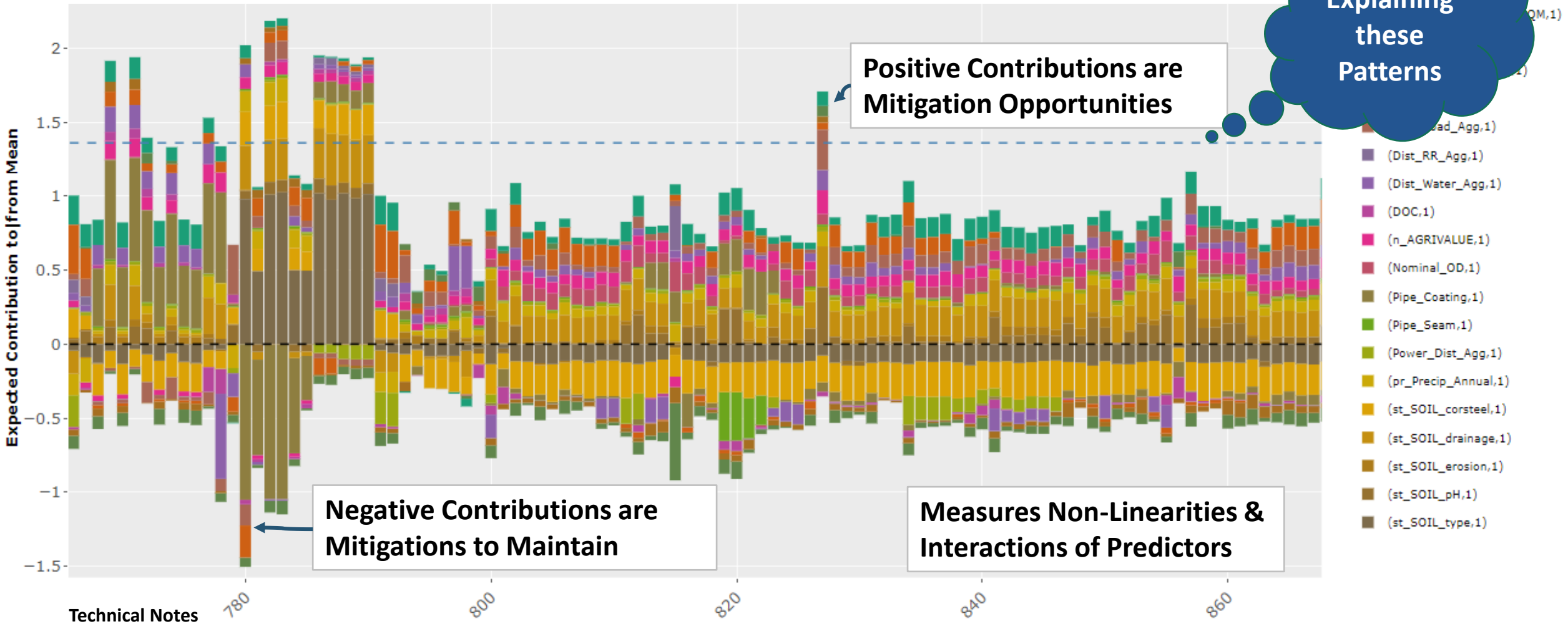
Predictor Values vs. Predictions

Technical Notes

- Critical Step in Domain Expert Review (Check Validity, Correlation, Causation)
- Importance Methods Reveal Predictor Influence at Global Level
- Considers Non-Linearities & Interactions
- Directionality based on Sampling of Actual Observations & Varying the Predictor Value
- Reveals Possible Monotonic Behavior

Explain Results

Predictor Contribution by Pipe Segment

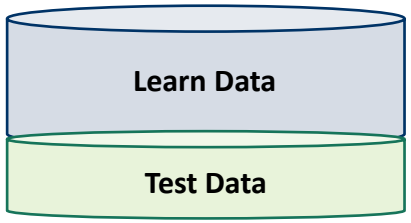


Technical Notes

- Deconstruction Methods are Either Model Dependent or Model-Agnostic
- Methods Deconstruct Predictions
- Variations in Contributions Consider Non-Linearities and Interactions
- Predictors may be Root Cause or Simply Correlated

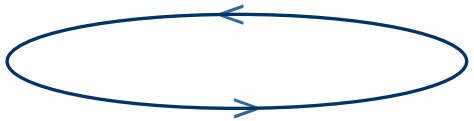
Machine Learning Process

Training Data
(Observations)



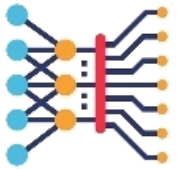
Training Data:
- Inspection Results
- Domain Expertise
- Incidents & Leaks
- Deterministic Results

Target of Interest
(Categorical\Numerical)



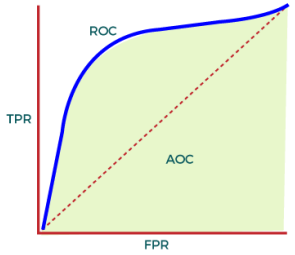
Learning Targets:
- Threats, Consequences
- Data Quality, Missing Data
- Program Costs
- Time to Criteria

Learned Model
(Target)



Models:
- Pattern Recognition
- Many Methods
- Model Tuning
- Predictive vs. Inference

Performance & Insights
(Validation & Acceptance)



Performance:
- Learning Curves & Unity Plots
- Uncertainty
- Model Acceptance
- Predictor Influence
- Directionality & Contribution

Technical Notes

Typical ML Processes
- Supervised (shown above)
- Unsupervised (no observations)
- Semi-Supervised
- Self-Supervised
- Synthetic Data Learning

Typical Targets
- Numerical (Regression)
- Two-Class (Classification)
- Multi-Class (Multi-Classification)

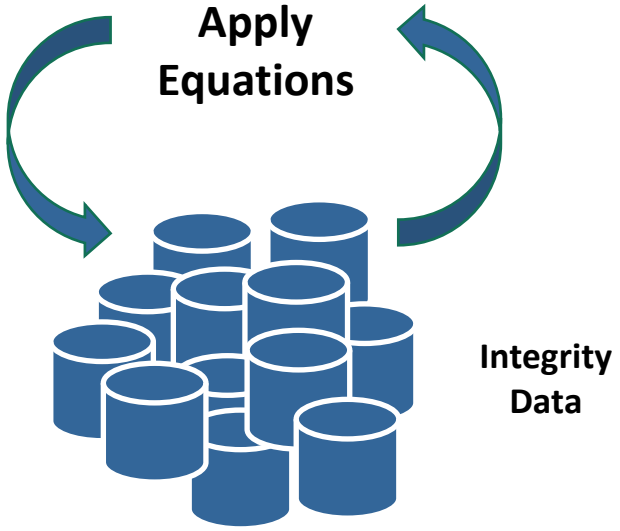
Models
- Hundreds of Methods
- Predictive
- Inferential (Explanatory)
- Ensembles

Typical Performance Metrics
- ROC, AUC, Accuracy
- Sensitivity, Specificity
- R2, RMSE, MAE
- KAPA, F1

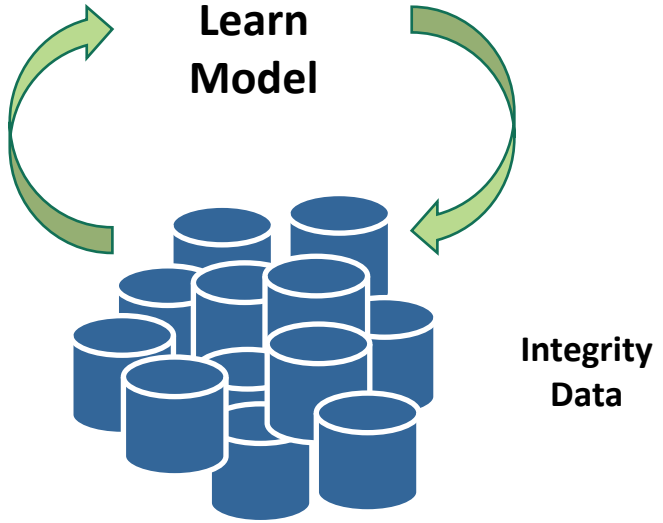
Differentiation - Deterministic vs. Learned

“Machine Learning Adapts Your Model To Your Business And Not The Business To Your Model”

Deterministic Models



Machine Learned Models



Same Integrity Data Used for Both Approaches

Unwanted Events are Complex

"The main idea behind complex systems (like pipeline integrity) is that the ensemble behaves in ways not predicted by its components. The interactions matter more than the nature of the units"

- Nassim Taleb, 2018 "*Skin in the Game*"

Technology

https://www.pipeline-risk.com

https://www.tidymodels.org/

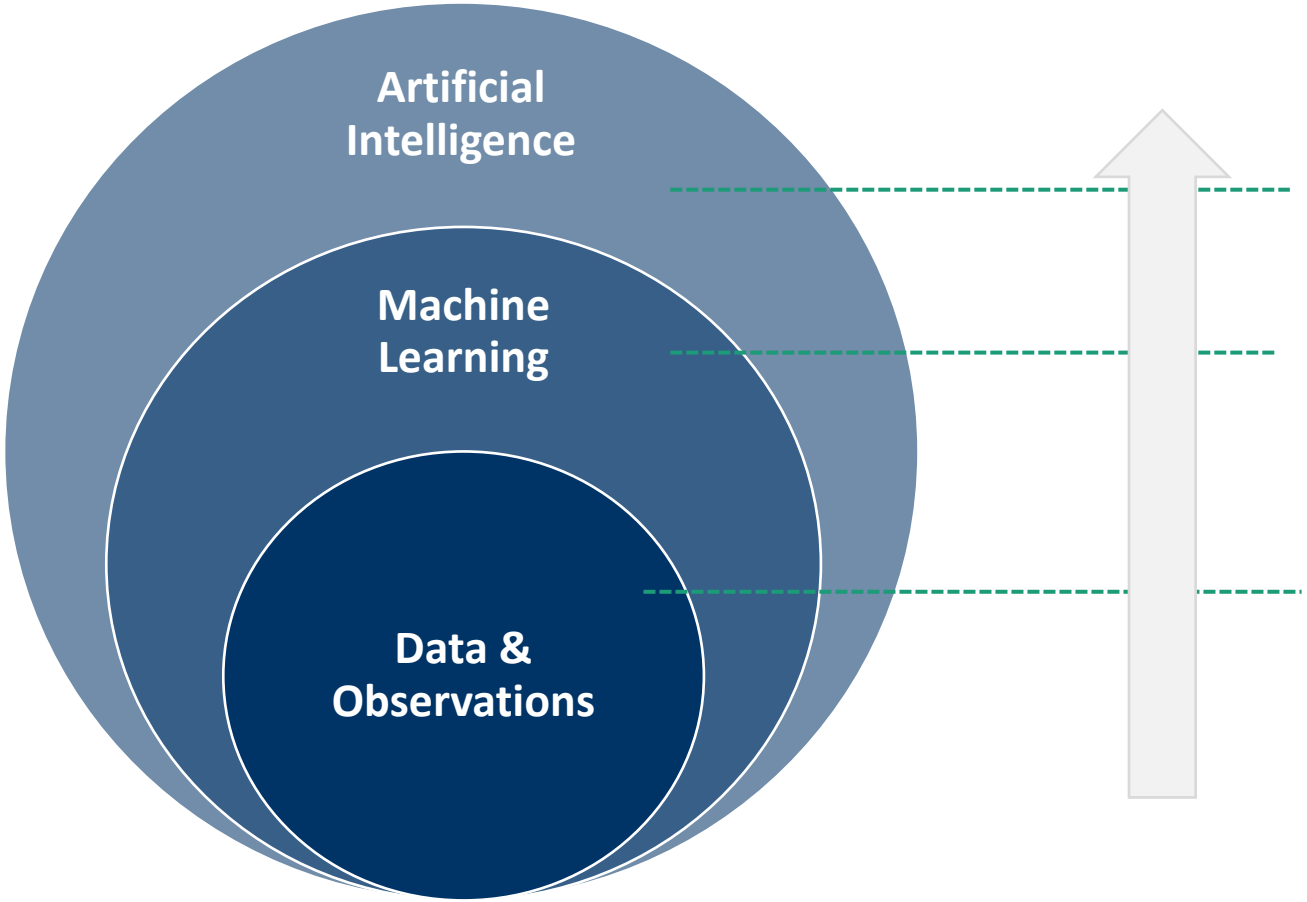
https://scikit-learn.org/stable/

Machine Learning Essentials

Unit 1.2

Common Questions

How Does Machine Learning Fit into AI



Automated Responses to Data Patterns (Adjust CP, Self-Repair, Reduce Pressure)

Identify Data Patterns Representing & Influencing External Corrosion Observations

External Corrosion **Observations** (Corrosion Defects) and Underlying Influencing Data (Coating Types, Soils, CP Readings, etc.)

Machine Learning – Common Questions

Topic	Discussion
Data	**Do I have enough data? Do I have the right data? How do you know?** • Use statistical tests and model performance to determine data adequacy • Consider using your existing risk data, assessment data, public data, industry base models as data sources
Complexity	**How can I interpret model mechanics and results?** • Consider using explanatory methods to explain models and results • These methods can deconstruct complex outputs into human readable results in support of risk mitigation analysis
Validation	**Are my current risk\threat models validated?** • Validate models with observational data, establish acceptance criteria, test with unseen data
Regulation	**Do regulators support approach?** • Consider PHMSA does not endorse any approach although they've been advocating better use of data and QRA • Note most of machine learning is based on statistical models and adoption in other related industries is strong
Readiness	**Are you Ready for ML?** • Consider you already have data you believe is important, and ML practices are mature and ready

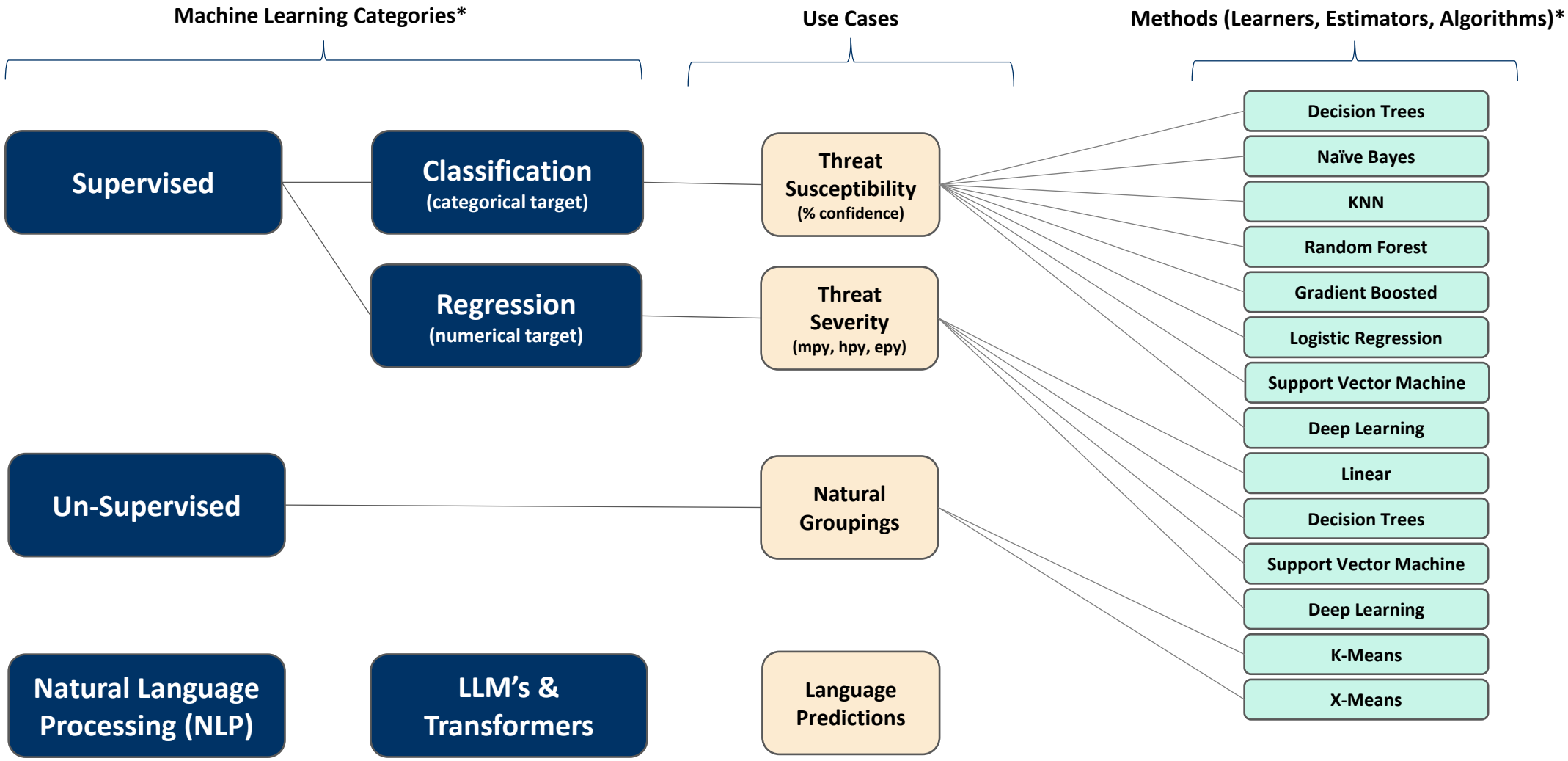
Machine Learning Essentials

Unit 1.3

Learning Categories

Learning Categories

Machine Learning Categories



* Generalized, Other Categories & Methods Exist

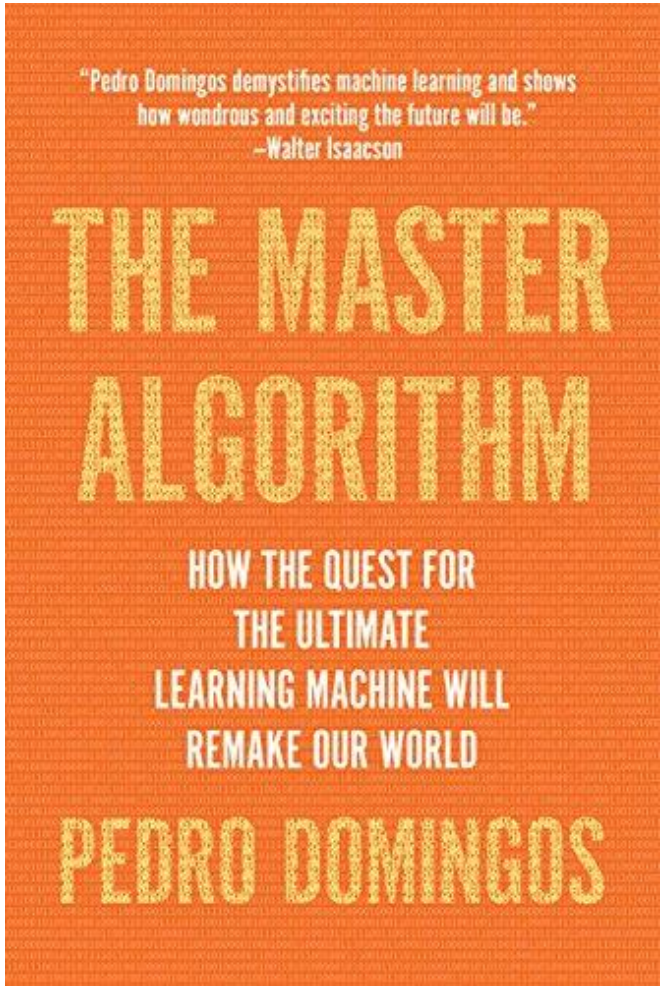
Machine Learning Essentials

Unit 1.4

Learning Methods

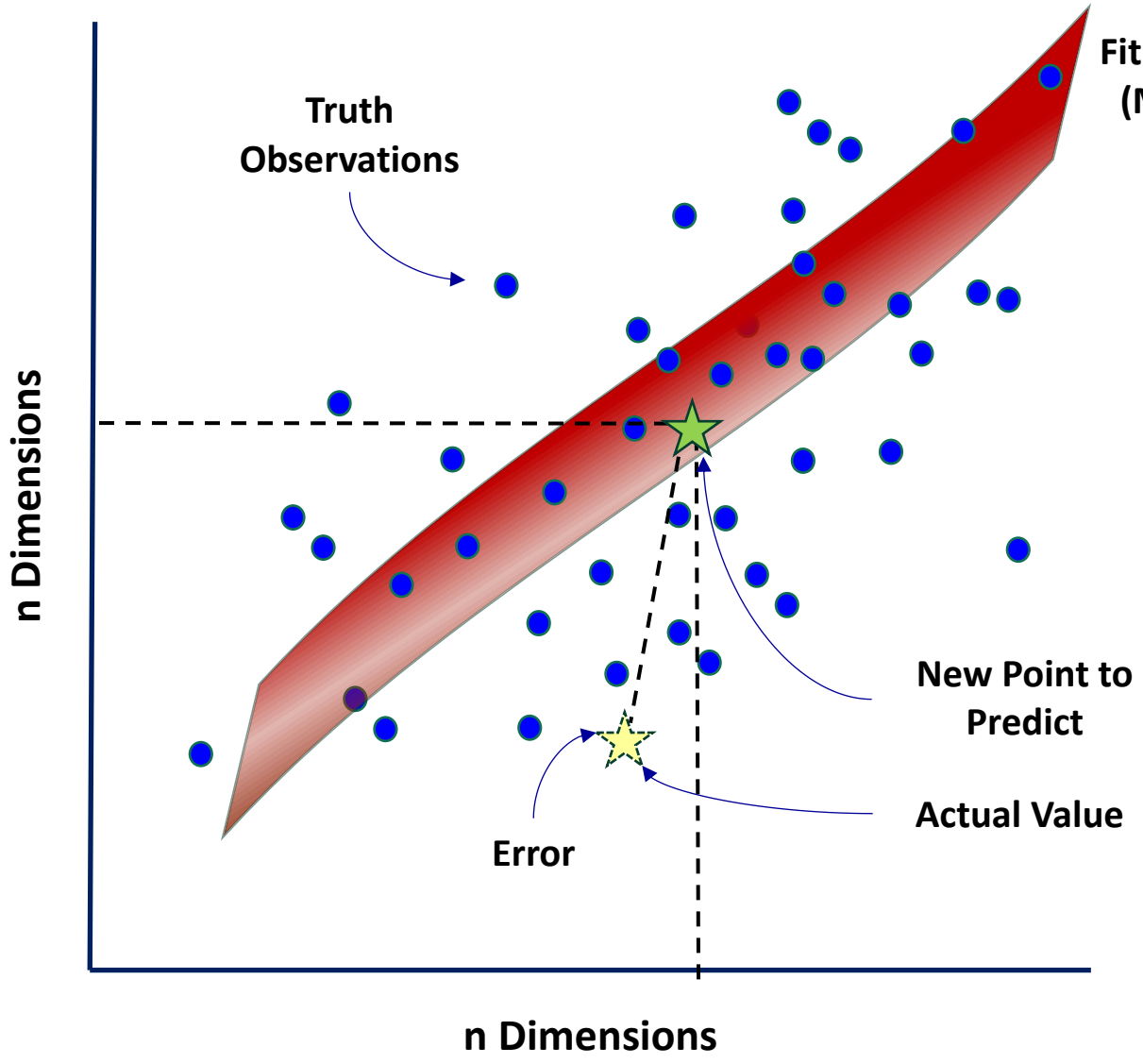
Learning Methods

- Many methods available to practitioner
- Choice depends on objectives (performance, cost, explainability, preference)
- Intent of machine learning process is to find best fitting model based on optimal coefficients and\or parameters of a method which meet your objectives
  - Coefficients - think linear regression y = mx + b
  - Parameters – think decision trees, number, depth, min breaks, distribution types, etc.



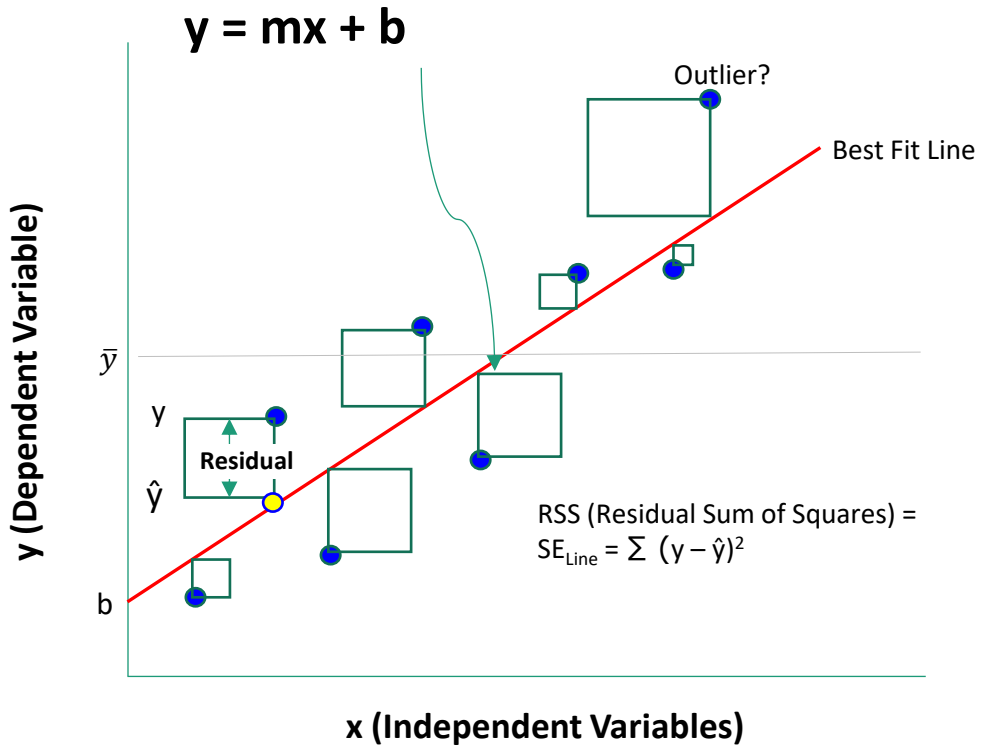
Common Models

Method Intuition – Finding Patterns in n-Dimensional Space



- Observations are Points in Vector Space
- Fit Pattern Regression or Classification
- Predict New Point
- Compare Prediction to Actual (Error)
- Is Prediction Error Acceptable?

Method - Linear Regression



Observation (x_n, y_n)

n = number of points, examples (m)

$\bar{y}$ = y mean

b = y intercept, bias

θ = coefficient, parameter, m (slope), weights

ŷ = prediction, h_θ (hypothesis function)

Error (residual) = y - ŷ , Unexplained error

Linear Regression involves finding a 'line of best fit' that represents a dataset using the least squares method. The least squares method involves finding a linear equation that minimizes the sum of squared residuals. A residual is equal to the actual minus predicted value.

1. **Find Best Fit Line**

   $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 \ldots\ldots \theta_j x_n$ (Hypothesis)

2. **Find θ's which minimize Squared Error of Line (SE_{Line})**

3. **Use Gradient Descent to Solve for θ Coefficients**

   Cost Function $J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta x^i - y^i\right)^2$

   Gradient Descent $\theta_j := \theta_j - \propto \frac{\partial}{\partial \theta_j} J(\theta)$

   where:

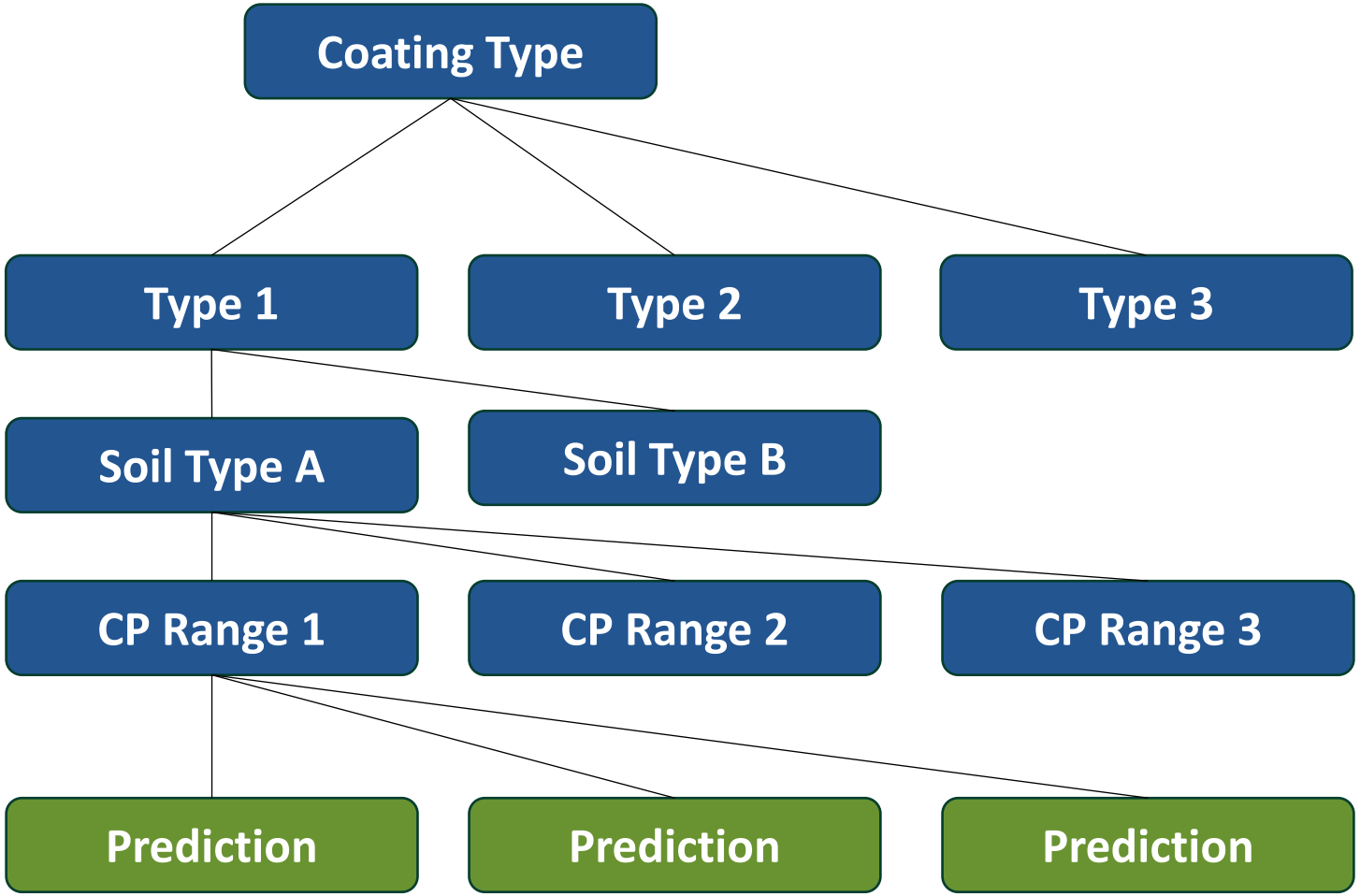
   θ_j are the coefficients (or m's) or weights to solve

   $h_\theta x^i$ is the hypothesis function

   $J(\theta)$ is the cost function to minimize

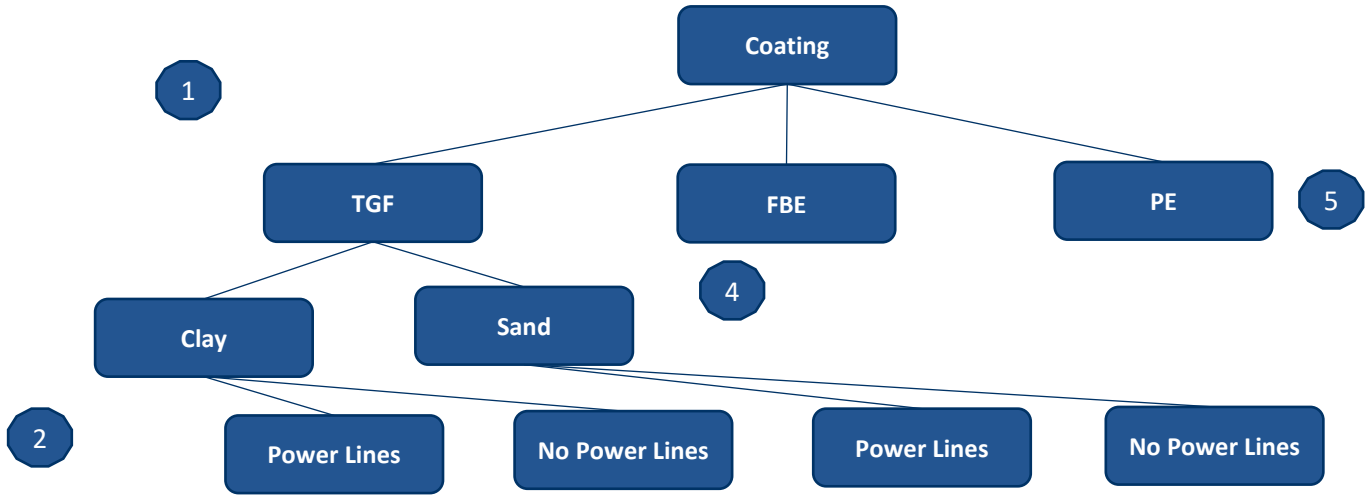
   $\propto$ is learning rate

Method Intuition – Finding Patterns using Trees



- Which Predictors Reduce Entropy the Most?
- Observations follow Path
- Predict New Point
- Compare Prediction to Actual (Error)
- Is Prediction Error Acceptable?

Method - Classification & Regression Decision Trees

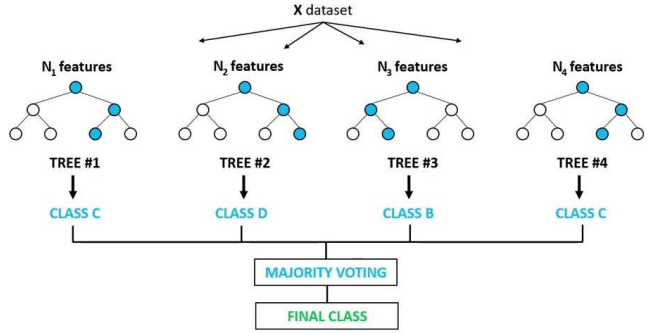


Avg Depth Corrosion	**.42**	**.10**	**.35**	**.17**

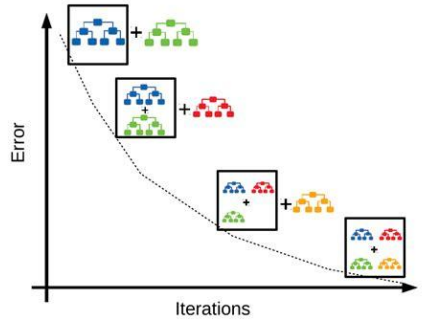
A **Decision Tree** is essentially a series of conditional statements that determine what path a sample takes until it reaches the bottom

- Creates Explicit Rules based on Observations
- Transparent, Easy to Interpret
- Can Handle Missing Data
- Under vs. Overfitting Issues

Bagging



Boosting

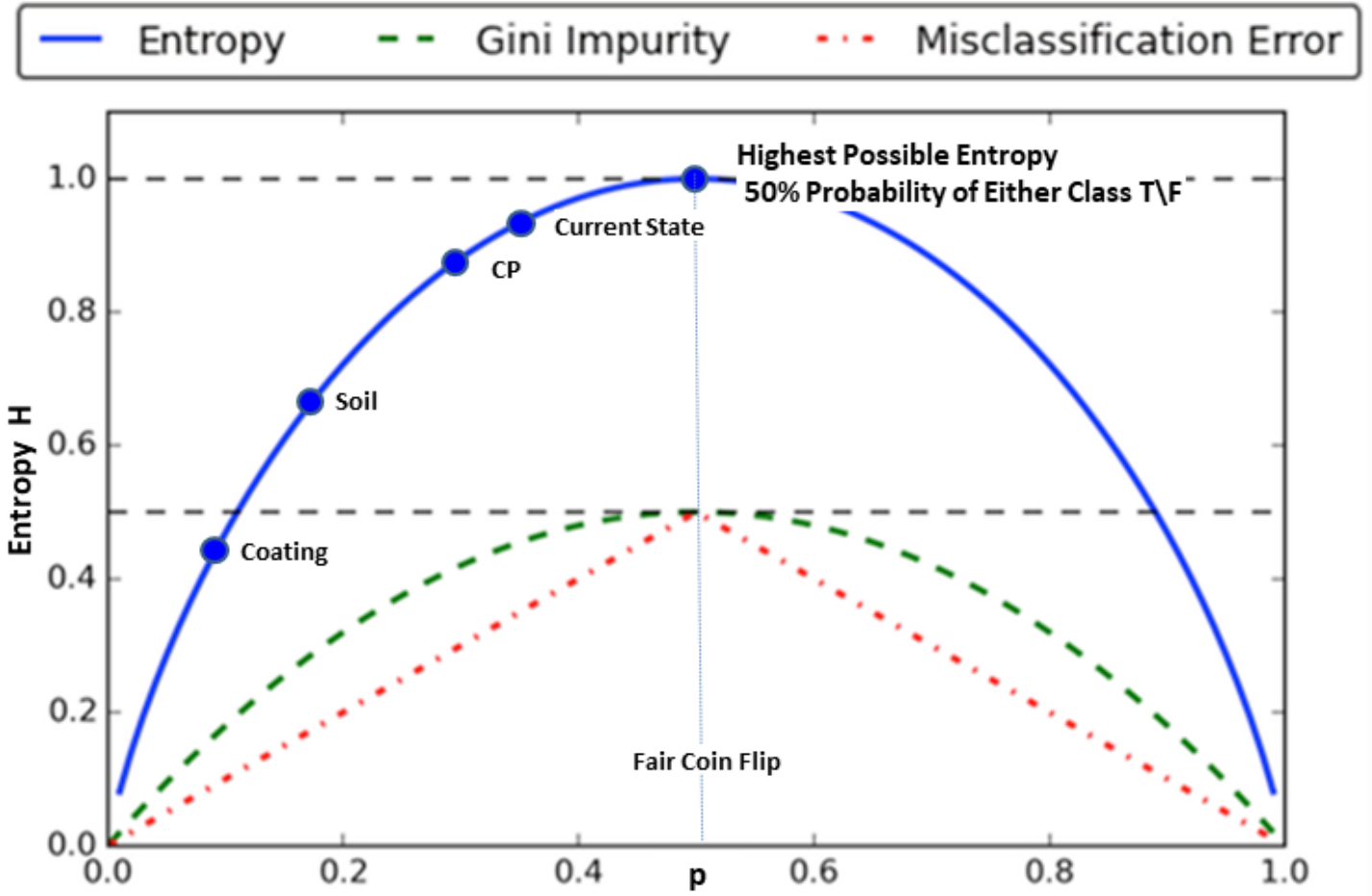


Parameters

1. Criterion – Splitting based on Selected Criteria
2. Depth – Maximum Tree Depth
3. Pre-Pruning (not shown) – Prune Nodes based on Criteria
4. Branch – Whether to Branch is based on Pruning Criteria
5. Leaf – Min\Max Leaf Sizes

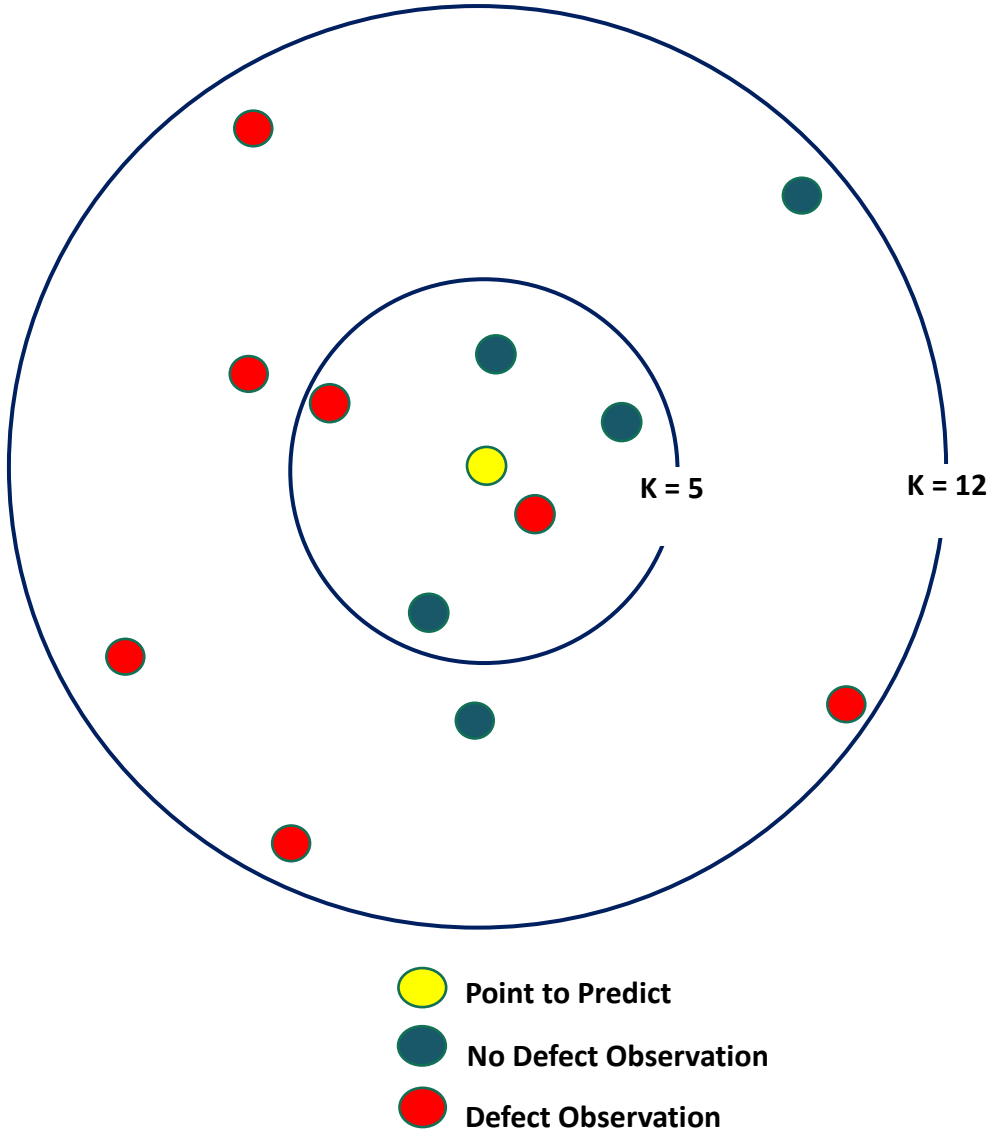
Entropy

Information Gain – Entropy Change



Shannon's Entropy H = $-p_a * \log_2(p_a) - p_b * \log_2(p_b)$

Method - KNN (K Nearest Neighbor)



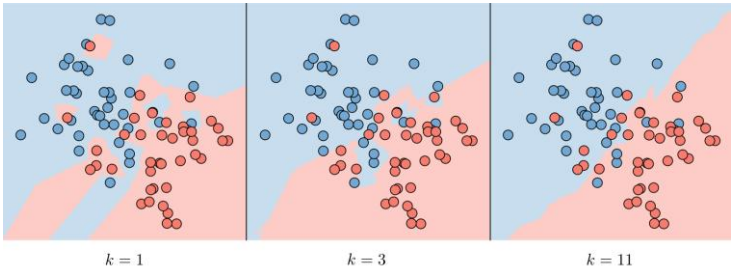
KNN

KNN Uses Distance Calculations to Predict Classifications

- Lazy Learner, Memorizes Training Data
- Shallow Learner
- Non-Parameterized Method (No Weightings)
- Numerous Distance Calculations Available
- Data Normalization Recommended
- User Enters “K” to get Prediction

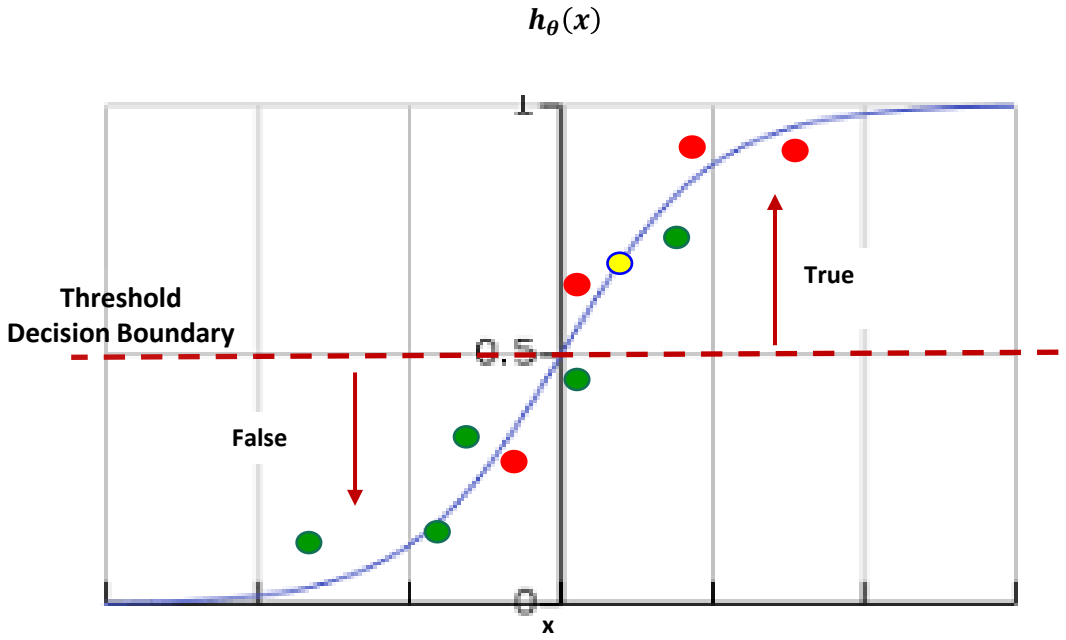
When K=5, the Prediction is “No Defect” (3 of 5, or 60% Confident)

When K=12, the Prediction is “Defect” (7 of 12, or 58% Confident)



The lower k (overfit), the higher the variance; the higher k (underfit), the higher the bias

Method - Logistic Regression



Defect Observation (x_n, y_n)

No Defect Observation (x_n, y_n)

ŷ = prediction, h_θ (hypothesis function), confidence

n = number of points, examples (m)

θ = coefficient, parameter, slope, weights

Logistic Regression is a classification technique that also finds a 'line of best fit.' However, unlike linear regression, where the line of best fit is found using least squares, logistic regression finds the line (logistic curve) of best fit using maximum likelihood. This is done because the y value can only be one or zero.

1. **Find Best Fit Line** $h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$ **(Hypothesis)**

- Sigmoid or Logistic Function for Binary Classification
- Linear Function $\theta^T X$ may also be replaced by a Nonlinear Function

2. **Use Gradient Descent to Solve for Coefficients**

Cost Function:

$$J(\theta) = -\frac{1}{m}\sum_{i=m}^{m}\left[y \cdot \log\left(h_\theta(x)\right) + (1-y) \cdot \log\left(1-h_\theta(x)\right)\right]$$

Gradient Descent $\theta_j := \theta_j - \propto \frac{\partial}{\partial \theta_j} J(\theta)$

where

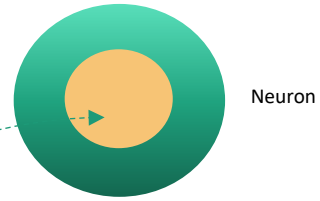
θ_j are the coefficients or weights to solve
$h_\theta x^i$ is the hypothesis function
$J(\theta)$ is the cost function to minimize
$\propto$ is learning rate

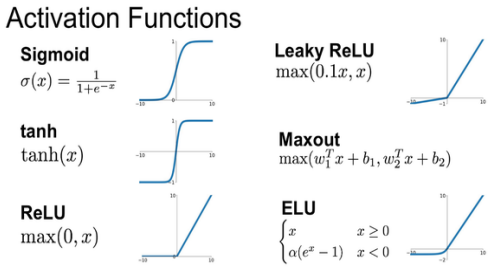
3. **Measure Performance thru Confusion Matrix**

Method - Deep Learning - Artificial Neural Network (ANN)

Some Types of ANN's (Nuclei)

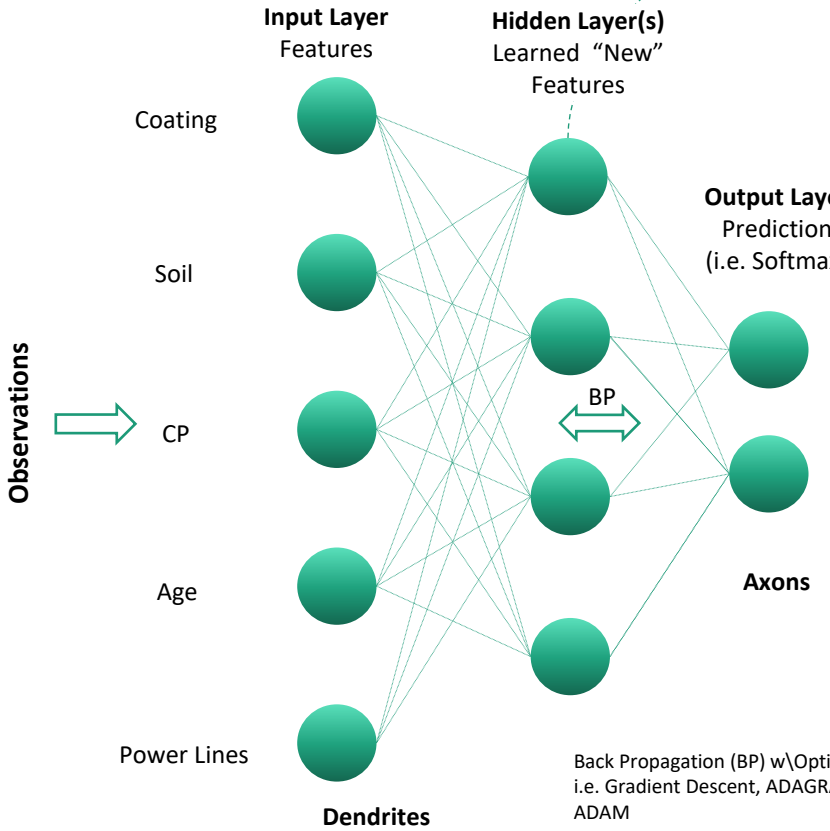
- MLP (Multi-Layer Perceptron) – Vanilla, Early Version
- RNN (Recurrent NN) – ST\LT Sequential Data, Speech
- CNN (Convolutional NN) – Vision
- DQN (Deep Q Network) – Reinforced Learning



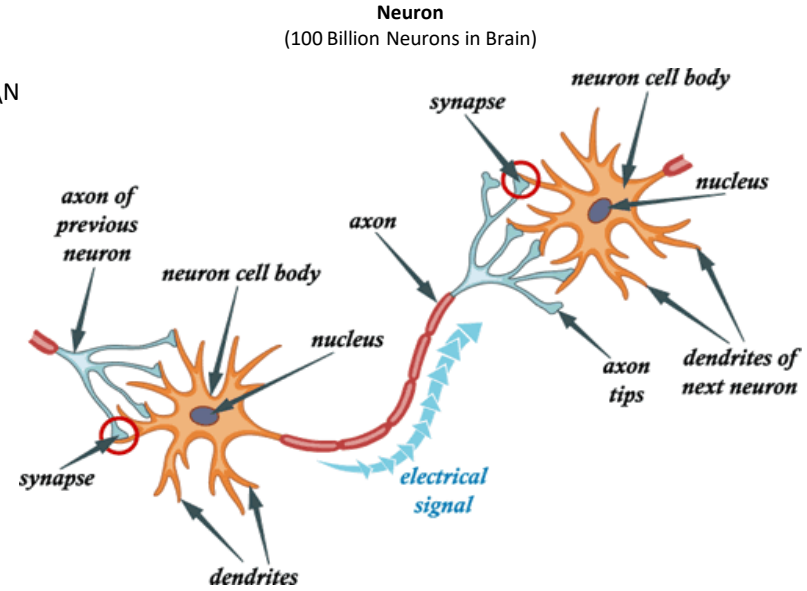


ANN

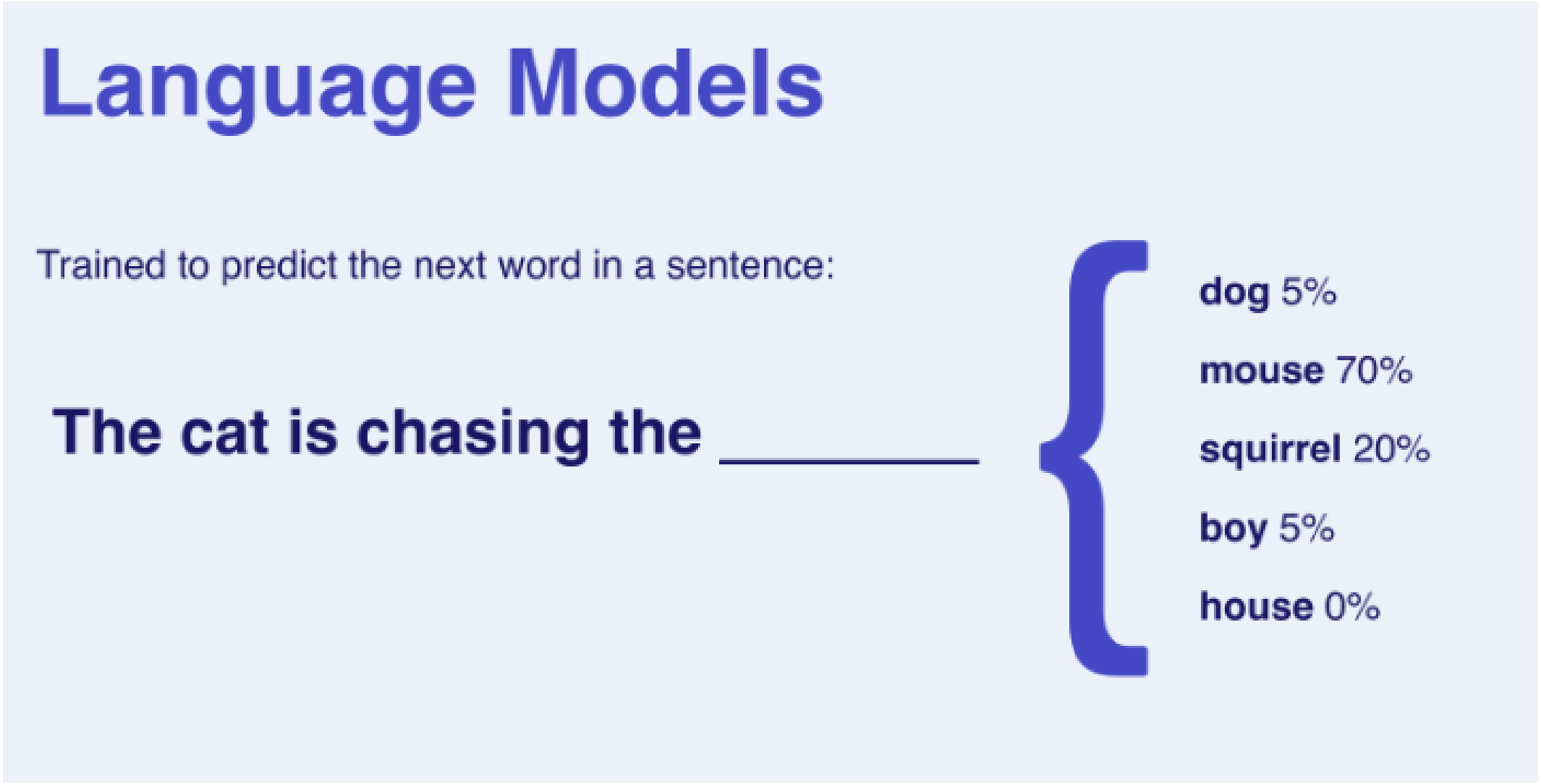
- Deep Learner
- Parameterized Method
- Higher Cost to Compute
- Linear or Non-Linear
- Data Normalization Recommended



- Corrosion Susceptibility Y\N
- Corrosion Severity mpy



Method – NLP's & Large Language Model (LLM's)



Summary

"All Models are Wrong, but Some Models are Useful"

- George E.P. Box
Famous Quality Control Mathematician

Machine Learning Essentials

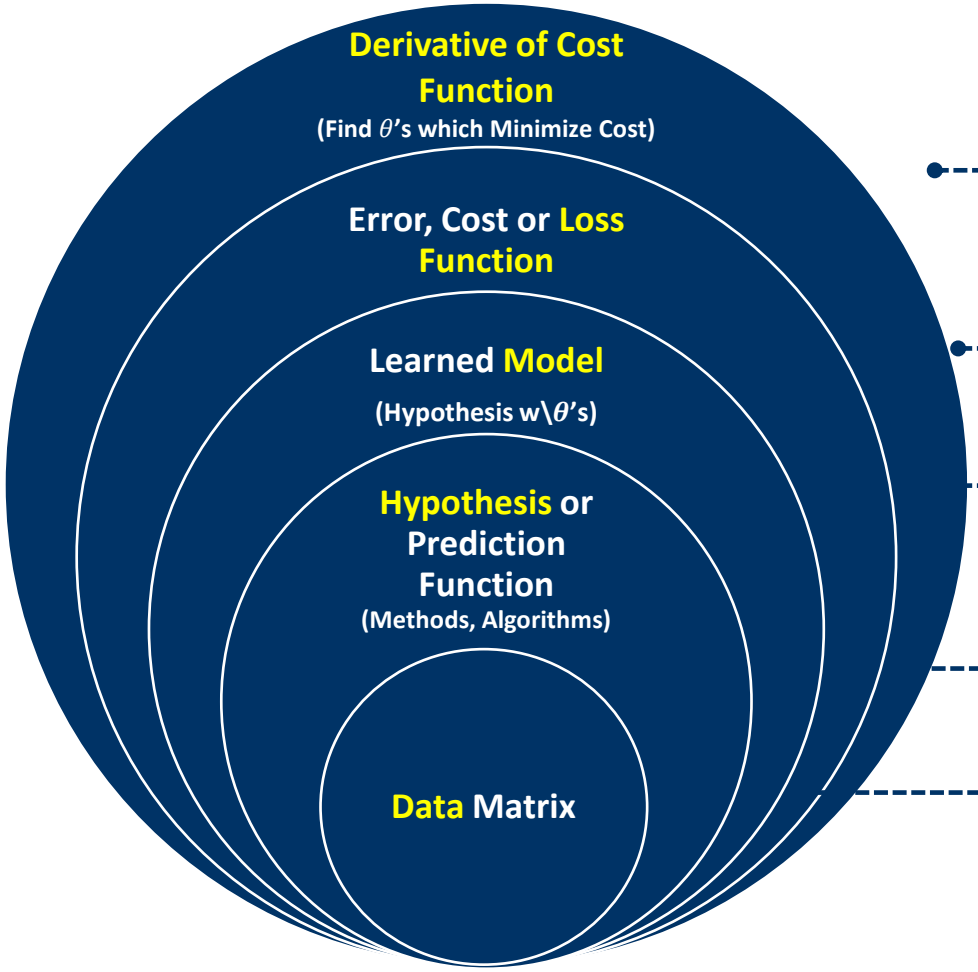
Unit 1.5

The Math

Gradient Descent

Where’s the bottom?

The Math – Gradient Descent



Regression
(ex. Linear Regression)

Gradient Descent (Derivative of Loss Function)*

$$\theta_j := \theta_j - \propto \frac{\partial}{\partial \theta_j} J(\theta)$$

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta x^i - y^i\right)^2$$

A Model with Learned Hypothesis Coefficients

$h_\theta(x) = \theta_0 + \theta_1 x^1 + \theta_2 x^2 \ldots\ldots \theta_p x^p$

Data
DynSeg n x k Matrix X
(Tensor)

* Other "Optimizer" functions are available

The Math – Regression Intuition

Prediction
Hypothesis model - this is what we want to predict

Unexplainable or Irreducible Error

Weights
A primary goal of Machine Learning is to determine the Theta's which lead to the "best" Model

$$h_\theta(x) = \text{Actual Value} \pm \varepsilon = \theta_0 + \theta_1 x_1 + \theta_2 x_2 \ldots\ldots \theta_j x_p \pm \varepsilon$$

Actual Observation

Bias
Can be considered bias if no other data is available

Feature Values

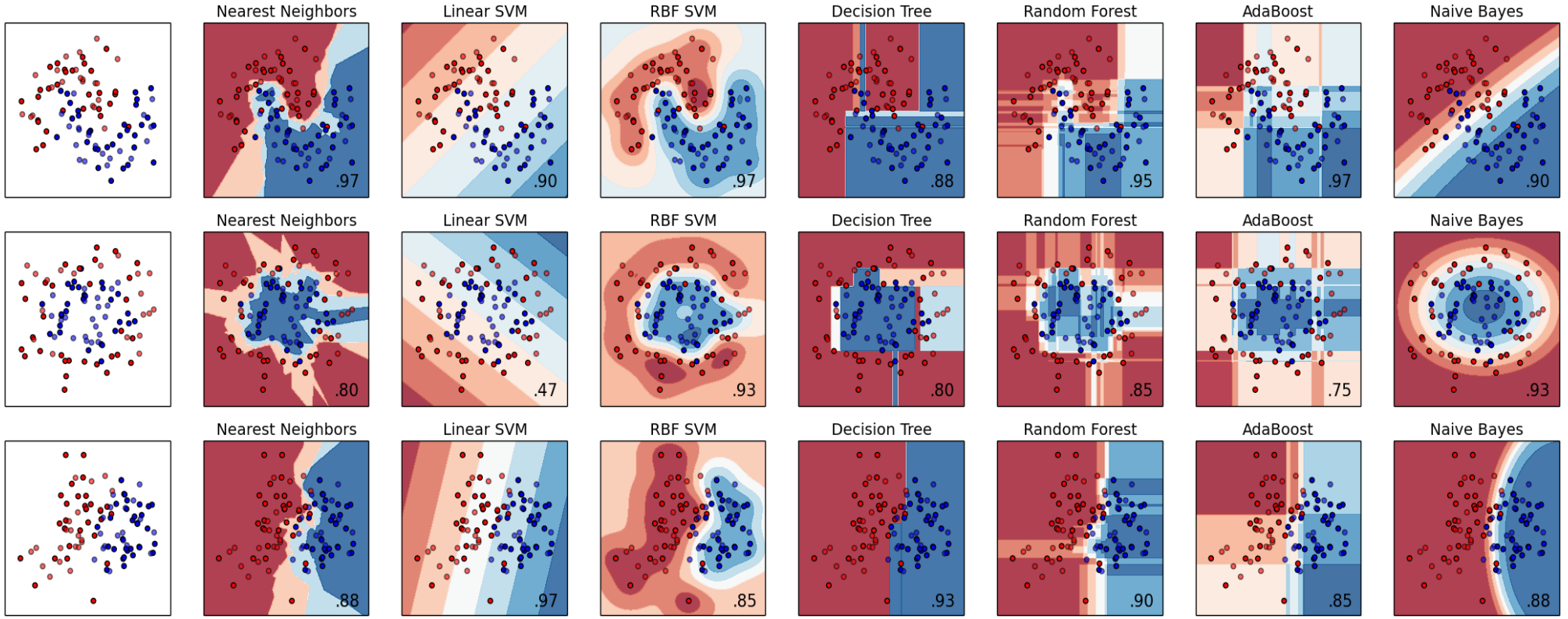
Machine Learning Essentials

Unit 1.6

Classification & Performance

Classification Methods

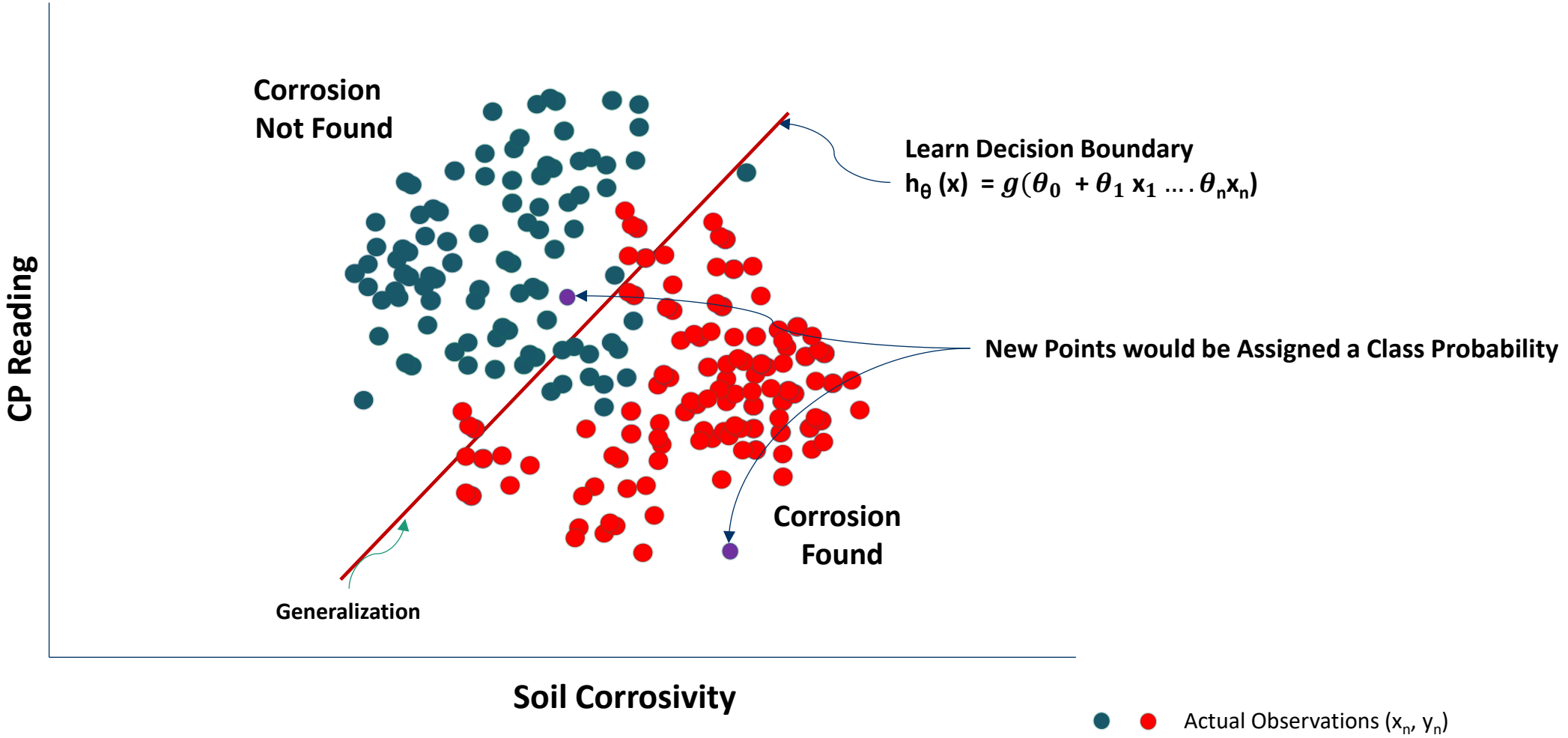
Example of Classifying Defects as Probability of Present or Not-Present in n-Dimensional Space



Example, red is “defect present”, blue is “defect not present”, number indicates accuracy

Intuition

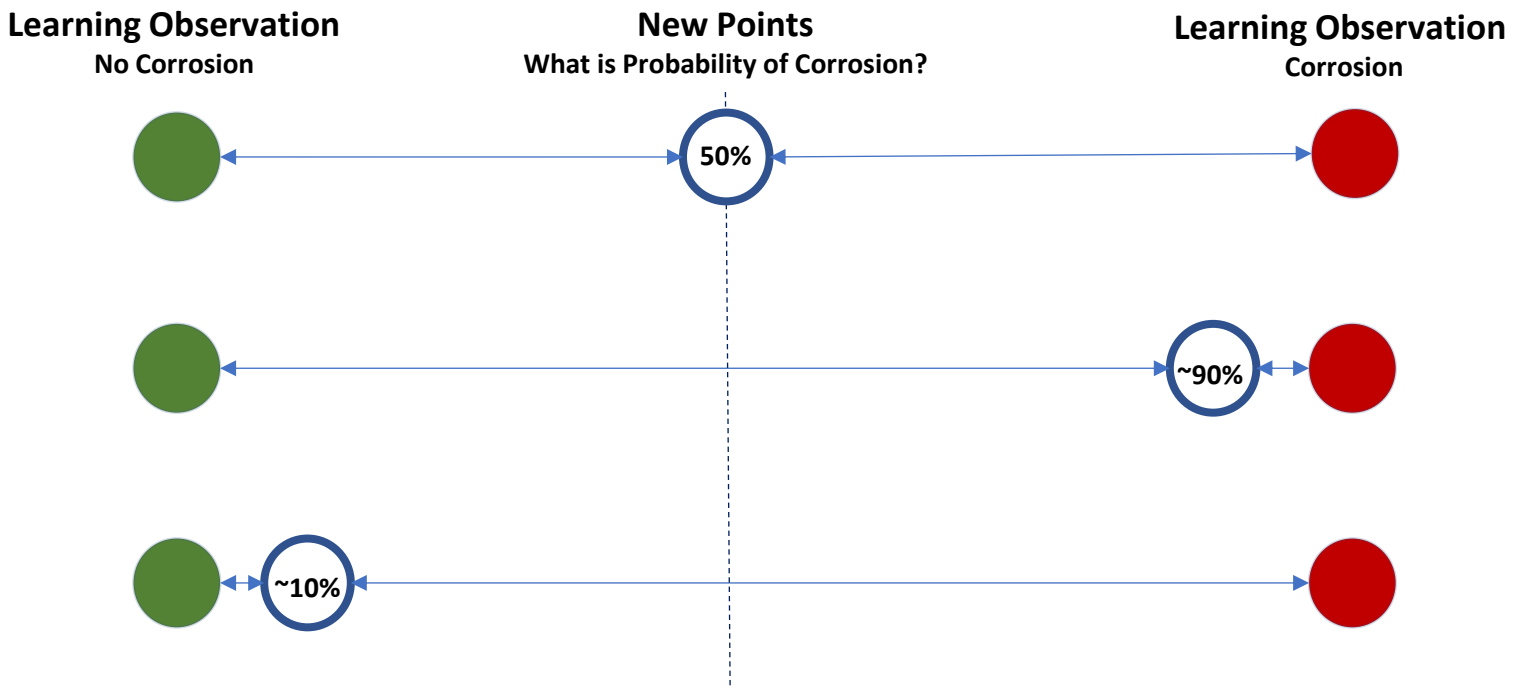
Plot Observations of External Corrosion Found vs. Known Variables



How does Classification "Classify" New Points?

Distances between Points in n-Dimensional Vector Space

(think Distance, Frequency & Variance)



No Corrosion <---- Threshold ----> Yes Corrosion

How do you make the class call?

Confusion Matrix

Two-Class Performance Learning Data Example:

- Joints of Pipe = 100
- Joints with Defects = 10
- Joints without Defects = 90

Overall Accuracy 89%	Actual (No Defects = 90)	Actual (Defects = 10)	
Prediction (No Defects = 81)	**80** (TN = true negatives)	**1** (FN = false negatives)	
Prediction (Defects = 19)	**10** (FP = false positives)	**9** (TP = true positives)	**47%** (precision)
	89% (specificity)	**90%** (sensitivity or recall)	

Classification Performance Metrics

Accuracy	(TP + TN)/(TP + TN + FP + FN)
Balanced Accuracy	(sensitivity + specificity)/2
Error	1 – Accuracy
Precision	TP/(TP + FP) = percentage of correctly predicted classes of predicted class (also positive predictive value)
Sensitivity	TP/(TP + FN) = percentage of correctly predicted classes of actual positive class
Specificity	TN/(TN + FP) = percentage of correctly predicted classes of actual negative class
False Positives	Type I Error
False Negatives	Type II Error
KAPPA	Useful metric when minority class is small (<.20 slight agreement, >.80 high agreement), higher KAPPA scores are better
Prevalence	(TP + FN)/(TP + TN + FP + FN)
F Score	2 * (precision * recall)/(precision + recall) = harmonic mean (higher F-Scores are better) = TP/(TP + ½(FP + FN))
Negative Predictive Values	TN/(TN + FN)
Detection Rate	TP/(TP + TN + FP + FN)
Detection Prevalence	(TP + FP)/(TP + TN + FP + FN)
LogLoss	Quantifies accuracy by penalizing false classification (smaller numbers better)

Machine Learning Essentials

Unit 1.7

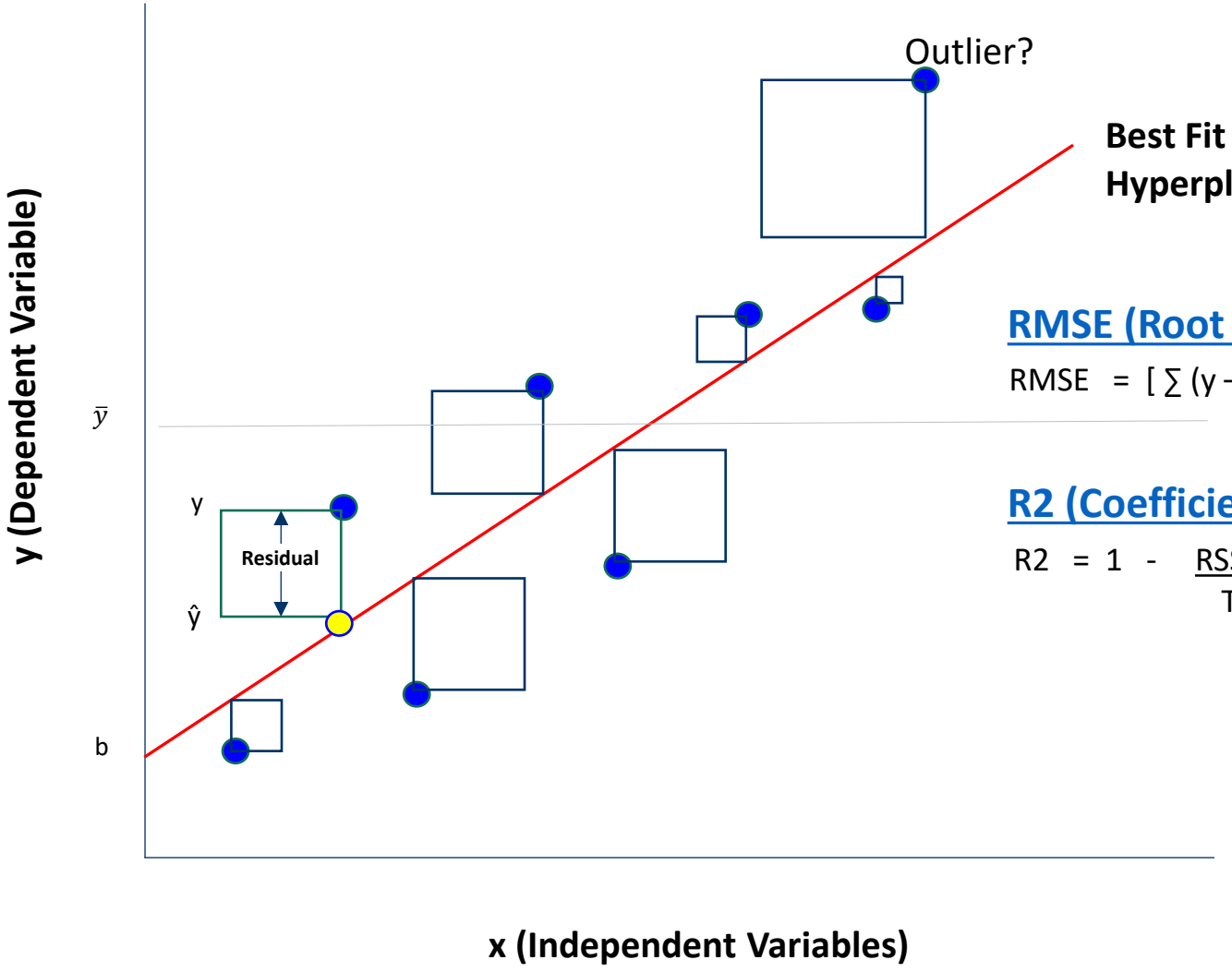
Classification Example

Machine Learning Essentials

Unit 1.8

Regression & Performance

Intuition



Cost Function will Minimize Error

RMSE (Root Mean Squared Error)

RMSE = $[\sum (y - \hat{y})^2 / n]^{½}$

R2 (Coefficient of Determination)

$$R2 = 1 - \frac{\text{RSS (Residual Sum of Squares)}}{\text{TSS (Total Sum of Squares)}}$$

Observation (x_n, y_n)

$\hat{y}$ = prediction, h_θ

b = y intercept, bias

$\bar{y}$ = y mean

n = number of points,

Machine Learning Essentials

Unit 1.9

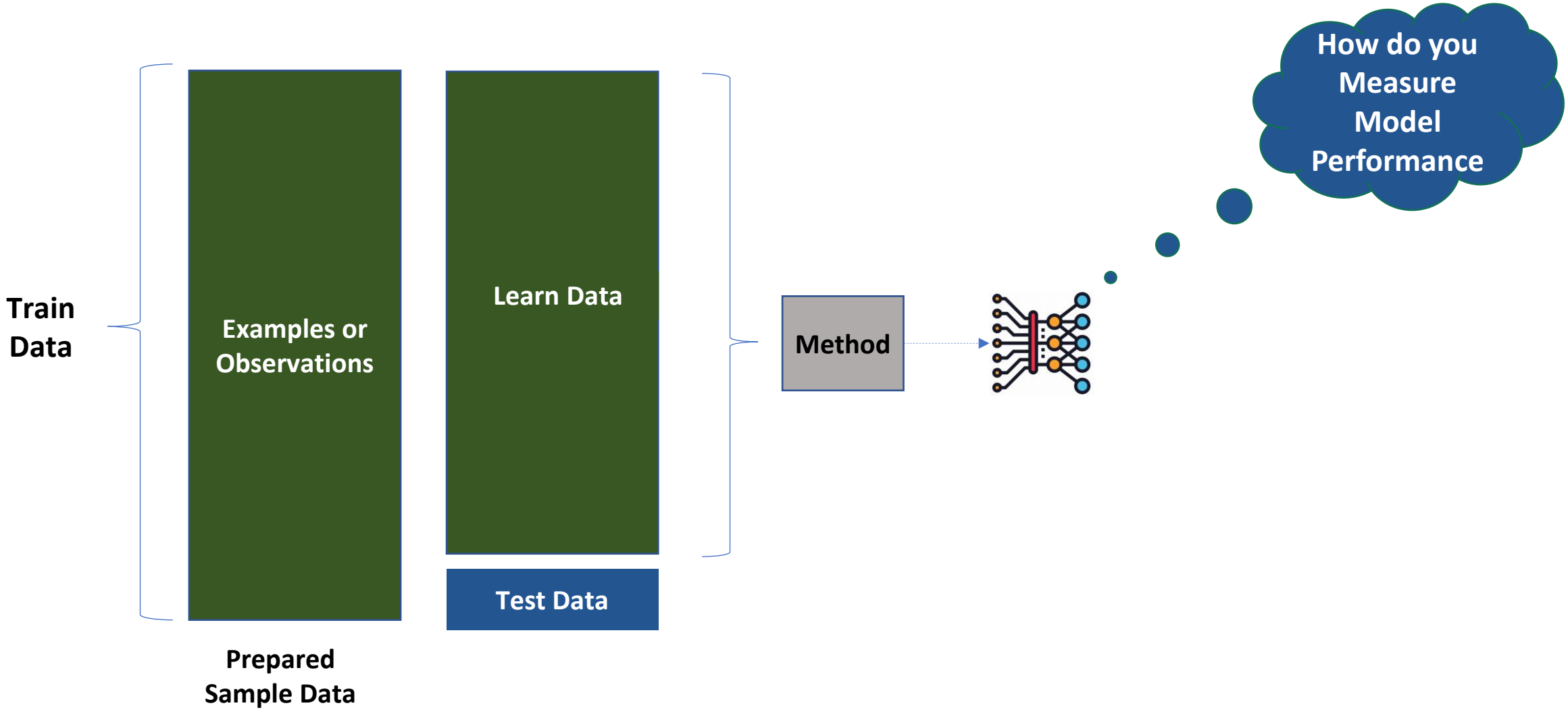
Regression Example

Machine Learning Essentials

Unit 1.10

Cross-Validation (Re-Sampling)

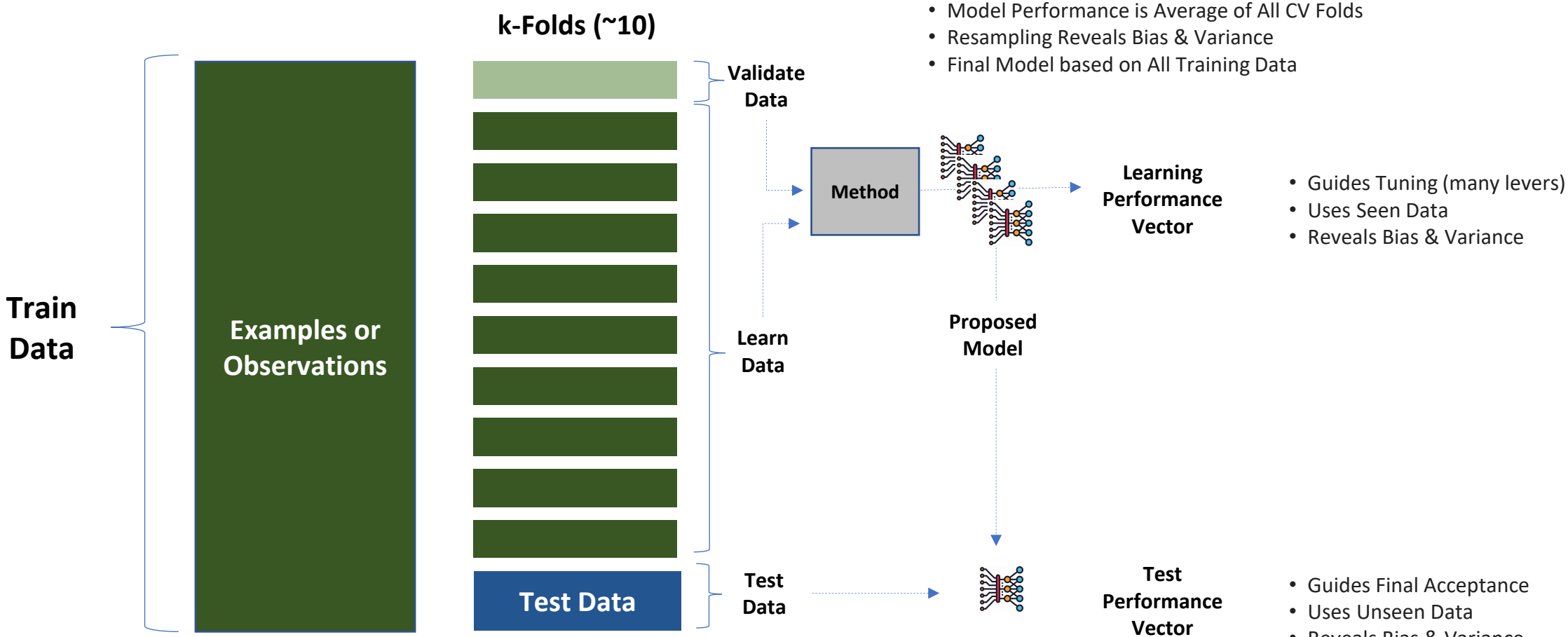
Why Resampling?



Cross-Validation & Resampling

Cross-Validation Allows for Description of Model Performance based on Learn Data

- Model Performance is Average of All CV Folds
- Resampling Reveals Bias & Variance
- Final Model based on All Training Data



- Guides Tuning (many levers)
- Uses Seen Data
- Reveals Bias & Variance

- Guides Final Acceptance
- Uses Unseen Data
- Reveals Bias & Variance

*Method Trained on 9 Sets & Validated on 1, Performed Iteratively 10 Times (k-Folds)

- A Strategy for Leveraging ALL Learning Data to Measure Performance and Learn a Final Model

Machine Learning Essentials

Unit 1.11

Model Explainability

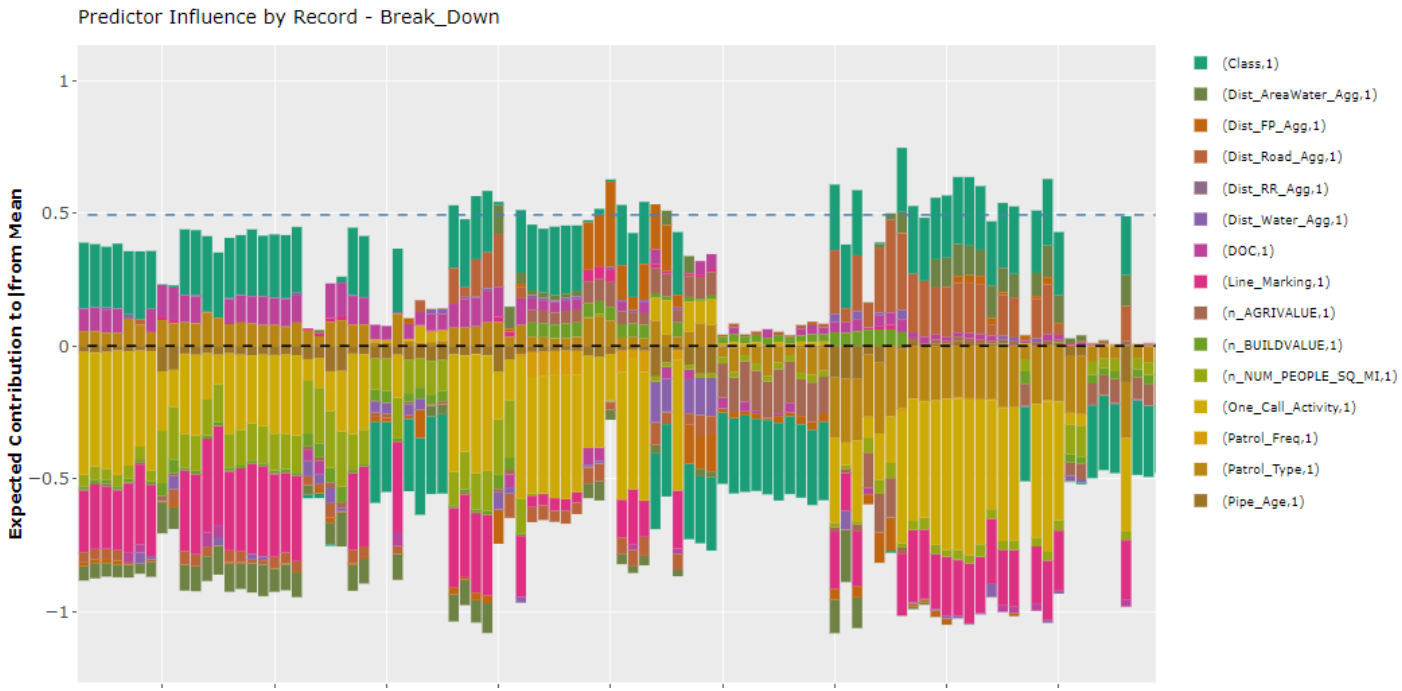
Model Explainability Methods

Objective

- Explain model predictions so they are understandable, explainable & validate results (i.e., 192.917)

Methods

- Global – generalizes explanation thru model weights and sampling of training data (explainer) results
- Local – explains specific prediction thru innovative statistical techniques
- Simulation – directly interact with model by changing inputs (predictors & legend)

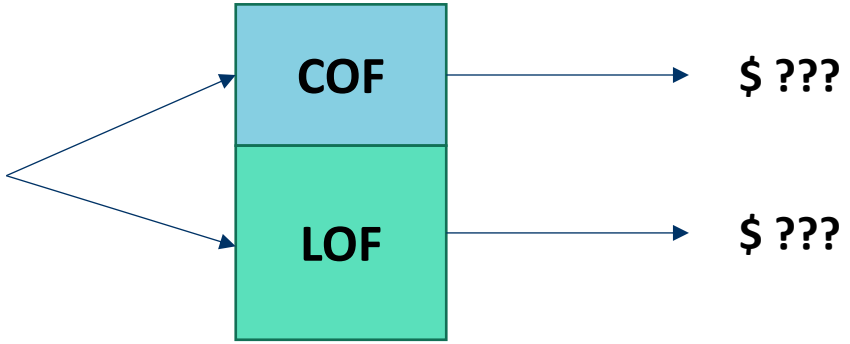


Intuition

ROF = LOF x COF

Example:

$120 = 60% x $200



- What is the Contribution of LOF to ROF in absolute terms?
- What is the Contribution of COF to ROF in absolute terms?

- Is it useful to know these contributions?
- What if you have a risk algorithm with 100 predictors & non-linearities?
- How do you know what each factor contributes?

Intuition

	Equations for Interactive Probabilities
P_{ECint}	$P_{ECint} = P_{EC} + 0.148(P_{EC} + P_{MFR}) + 0.017(P_{EC} + P_{CD}) + 0.013(P_{EC} + P_{MD})$
P_{ICint}	$P_{ICint} = P_{IC} + 0.052(P_{IC} + P_{MFR}) + 0.019(P_{IC} + P_{CD})$
P_{SCCint}	$P_{SCCint} = P_{SCC}$
P_{MFRint}^{*}	$P_{MFRint} = P_{MFR}$
P_{CDint}	$P_{CDint} = P_{CD}$
P_{EFint}	$P_{EFint} = P_{EF} + 0.059(P_{EF} + P_{EC})$
P_{MDint}	$P_{MDint} = P_{MD} + 0.089(P_{MD} + P_{SCC}) + 0.022(P_{MD} + P_{CD}) + 0.006(P_{MD} + P_{IO})$
P_{IOint}	$P_{IOint} = P_{OI} + 0.101(P_{IO} + P_{EF}) + 0.055(P_{IO} + P_{EC}) + 0.023(P_{IO} + P_{CD})$
P_{OFint}	$P_{OFint} = P_{OF} + 0.092(P_{OF} + P_{EF}) + 0.203(P_{OF} + P_{CD}) + 0.098(P_{OF} + P_{MD})$

Machine Learning Essentials

Unit 1.12

Explainability Example

DATA

Data Concepts

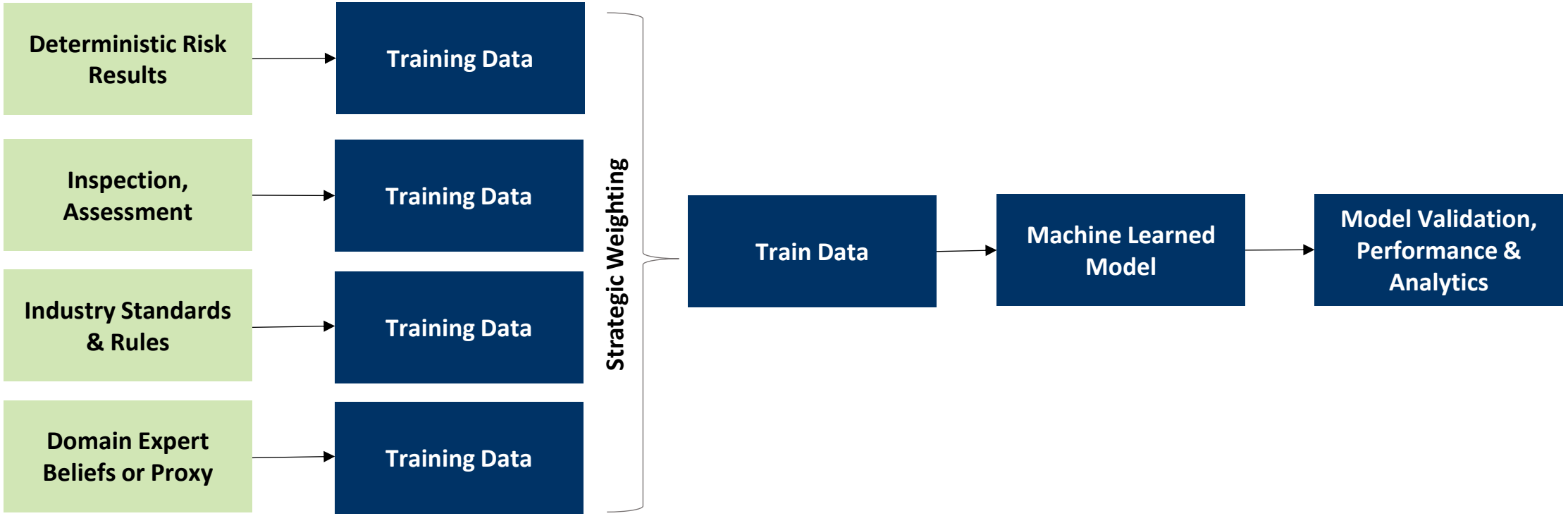
Unit 2.1

Training Data

Training Data Sources

Categories

- The Pipe
- Pipe Operation & Performance
- Outside the Pipe

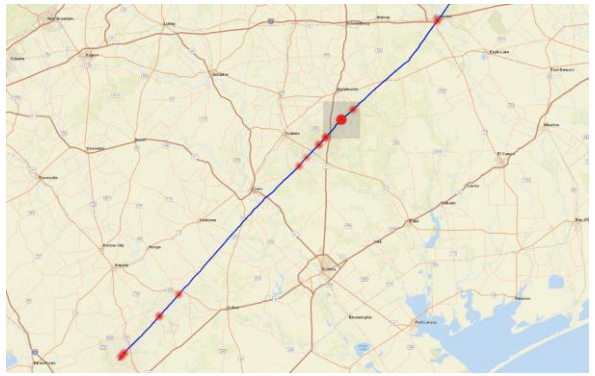


Data Concepts

Unit 2.2

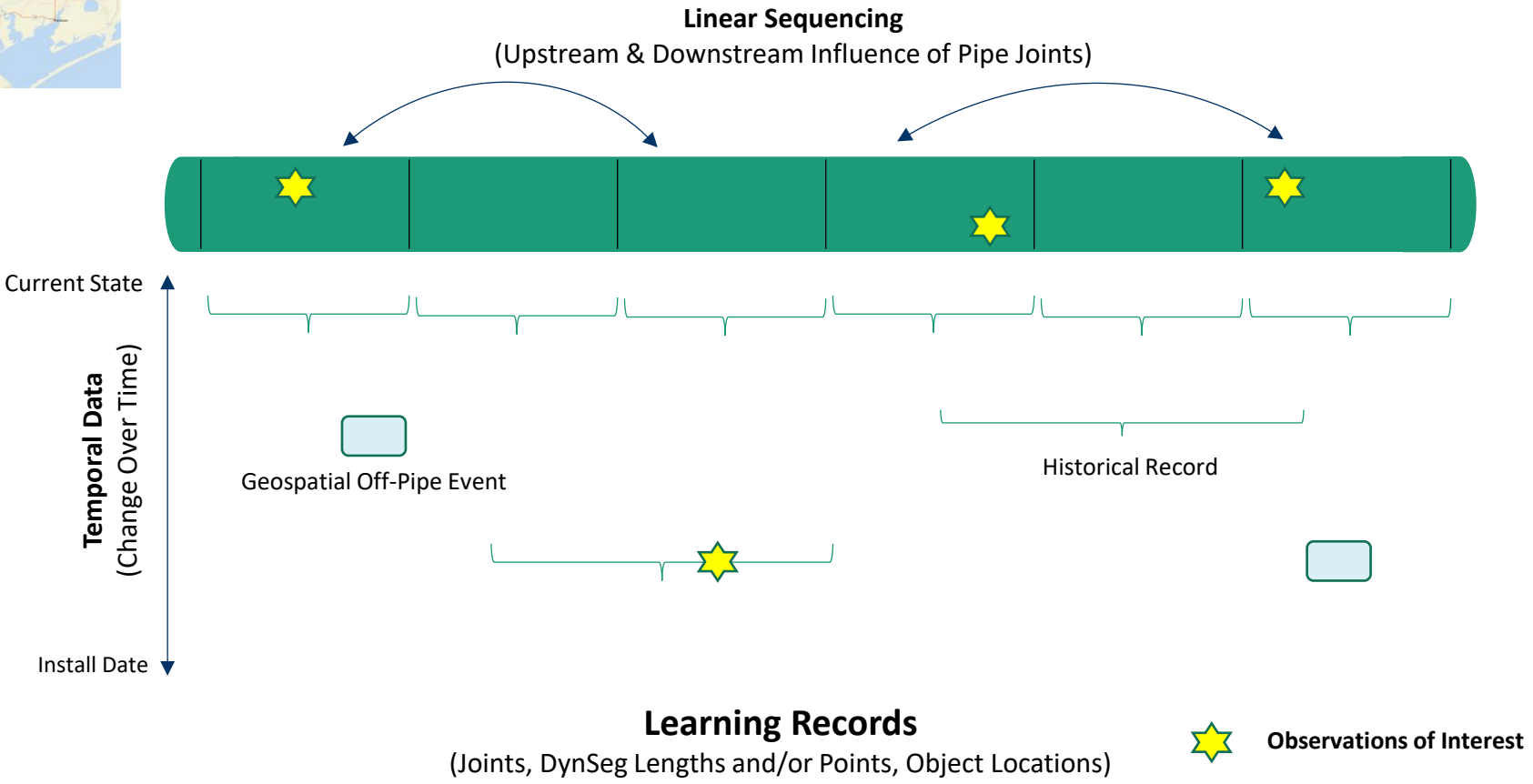
Data Integration

What's Special About Pipeline Data?

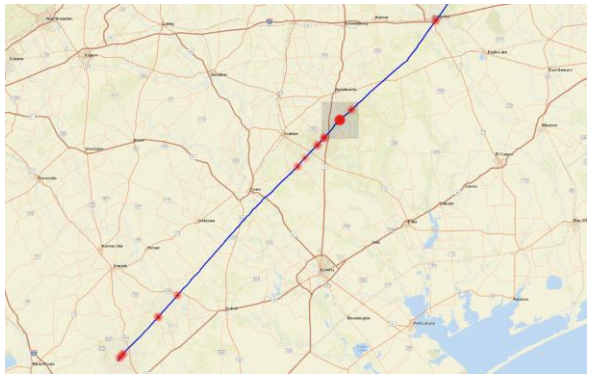


Dimensions

- Segmentation (Use Joints)
- Temporal (Use Rates – Feature Engineering)
- Off-Pipe Events (Use Thresholds)
- Sequencing (Interpolate, Generalize)



External Corrosion Example



Linear Sequencing
(Upstream & Downstream Influence of Pipe Joints)

- **Corrosion Volume**
- **CP Interpolation**

Current State

Temporal Data
(Change Over Time)

- **Pipe Joints**
- **Local Observations**

Geospatial Off-Pipe Event

Historical Record

- **CP Change**
- **Rectifier Change**
- **Past Repairs**

- **HVAC**
- **Foreign Lines**

Install Date

Learning Records
(Joints, DynSeg Lengths and/or Points, Object Locations)

Observations of Interest

Resulting Training Data - External Corrosion Example

	Target	Pipe Segmentation					Temporal		Sequence	Off-Pipe			
Comp_Name	EC_MPY	Measure_Start	Measure_End	Coating	Long_Seam	SOIL_PH	CP_Off_Trend	ANN.TEMP_DIFF	Comp_Distance	OHPL.Crossing.Distance	SCC_BINARY	TPD_BINARY
All	All	All	All	All	All	All	All	All	All	All	All	All
Pipeline_3	0.32	112	611	TGF_E	DSAW	5.80	-0.02	20.60	112.00	10,000.00	No	No
Pipeline_3	0.24	611	1142	TGF_E	DSAW	5.80	-0.02	20.60	611.00	10,000.00	No	No
Pipeline_3	0.25	1142	1650	TGF_E	DSAW	5.80	-0.00	20.60	1,142.00	10,000.00	No	No
Pipeline_3	0.19	1650	2146	TGF_E	DSAW	5.80	-0.02	20.60	1,650.00	10,000.00	No	No
Pipeline_3	0.27	2146	2629	TGF_E	DSAW	5.80	-0.00	20.60	2,146.00	10,000.00	No	No
Pipeline_3	0.20	2629	3119	TGF_E	DSAW	5.80	-0.01	20.60	2,629.00	10,000.00	No	No
Pipeline_3	0.00	3119	3660	TGF_E	DSAW	6.50	-0.00	20.60	3,119.00	10,000.00	No	No
Pipeline_3	0.00	3660	4120	TGF_F	DSAW	6.50	-0.01	20.60	3,660.00	10,000.00	No	No
Pipeline_3	0.32	4120	4638	TGF_E	DSAW	6.50	-0.01	20.60	4,120.00	10,000.00	No	No
Pipeline_3	0.00	4638	5160	TGF_F	DSAW	6.50	-0.00	20.60	4,638.00	10,000.00	No	No
Pipeline_3	0.00	5160	5607	TGF_F	DSAW	6.50	0.02	20.60	5,160.00	10,000.00	No	No
Pipeline_3	0.27	5607	6110	TGF_E	DSAW	6.50	0.00	20.60	5,607.00	10,000.00	No	No
Pipeline_3	1.00	6110	6650	TGF_E	DSAW	6.50	0.01	20.60	6,110.00	10,000.00	No	No
Pipeline_3	0.00	6650	7157	TGF_E	DSAW	6.50	0.01	20.60	6,650.00	10,000.00	No	No
Pipeline_3	0.27	7157	7598	TGF_E	DSAW	6.50	0.01	20.60	7,157.00	10,000.00	No	No

Data Concepts

Recommendations

- Understand your objective (learning target)
- Understand your data
- Leverage existing & public data

Challenges

- Do I have enough data? How do I know? What if I don’t?
- Do I have the right data? ? How do I know? What if I don’t?

Concepts

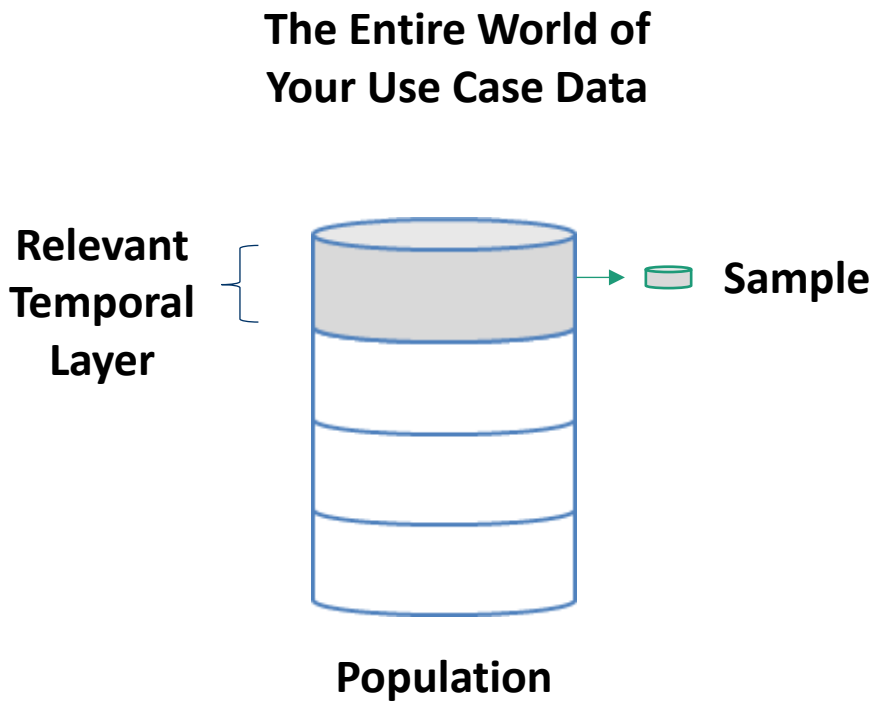
- Use data proxies
- Use synthetic data (human or machine generated)
- Use machine learning to measure how much and what data
- Learning data tends to be more granular
- Prediction data tends to be less granular

Data Concepts

Unit 2.3

Data Sampling

How Much Learning Data?

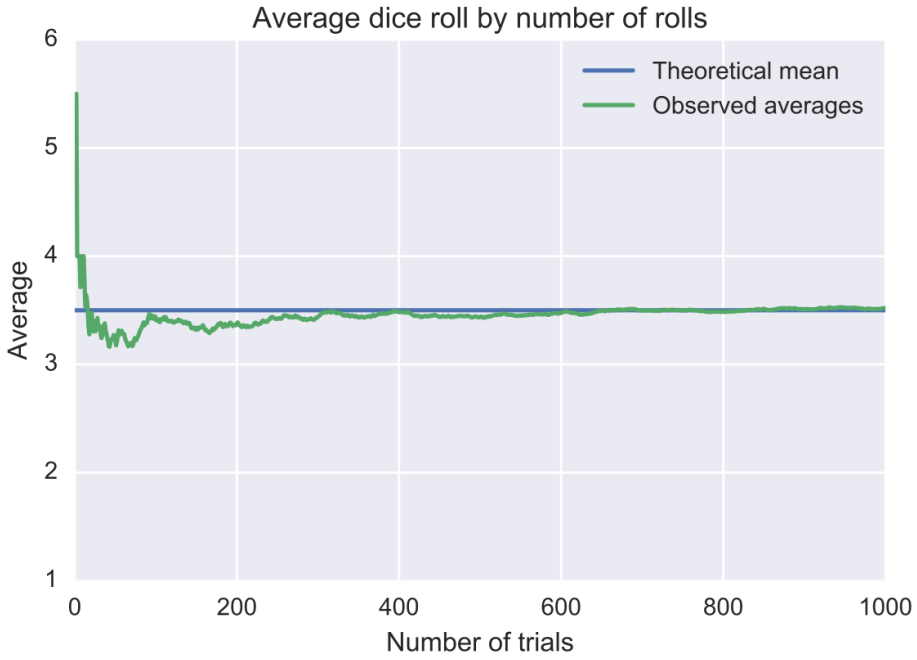


The Learning Sample

- Samples should be a randomly selected set of records from the entire population of interest
- Verified thru analysis & visualization by domain experts
- Apply hypothesis testing and other methods to diagnose and mitigate sampling errors
- Performance metrics of learned models may reveal sampling errors

Law of Large Numbers & Central Limit Theorem

- The **Law of Large Numbers** states that the average of the results obtained from many trials tends to become closer to the population average as more trials are performed



States that Given A Sufficiently Large Sample

- The means of the samples in a set of samples (the sample means) will be approximately normally distributed
- This normal distribution will have a mean close to the mean of the population

Why Do We Care?

- Most if not all data we use for integrity & risk analysis is a sample of a larger population, and we can use the CLT to infer unknown population parameters and confidence intervals

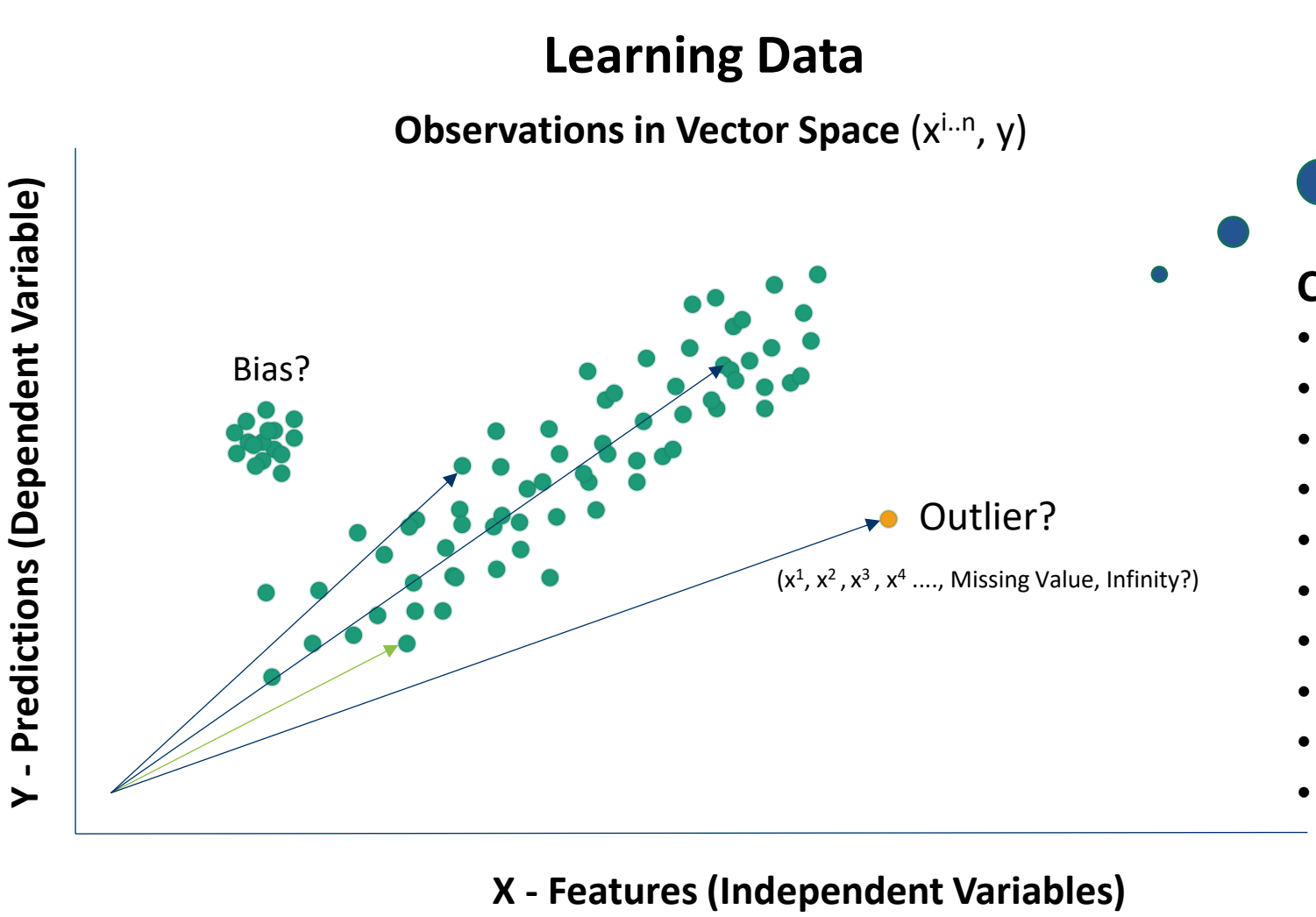
https://onlinestatbook.com/stat_sim/sampling_dist/index.html

Data Concepts

Unit 2.4

Data Quality

Intuition - What Does Data Quality Mean?



Does my Data Match Reality?

Concepts

- Does Sample Represent Population?
- Is there Missing Data?
- Are there Long Factor\Attribute Lists?
- Are there Correlations & Confounders?
- Are there Outliers?
- Are there Natural Clusters, Bias?
- Are there Similarity Issues
- Is Referencing Correct?
- Are there Temporal Issues?
- Is Data Just Incorrect?

Data Concepts

Unit 2.5

Learning Data Pre-Processing

Why is Training Data Pre-Processed?

Some Concepts (Objective is to Improve Model Performance)

- ❑ Puts Predictor Data on Same Scales (number of standard deviations)
- ❑ Converts (Sometimes) Categorical Data to Numerical (0/1)
- ❑ Imputes Missing Data
- ❑ Upsamples Minority Data
- ❑ Removes Highly Correlated Data
- ❑ Engineers Features to Improve Performance (PCA)

Pre-Processing Recipes

A Structured Approach to Data Pre-Processing

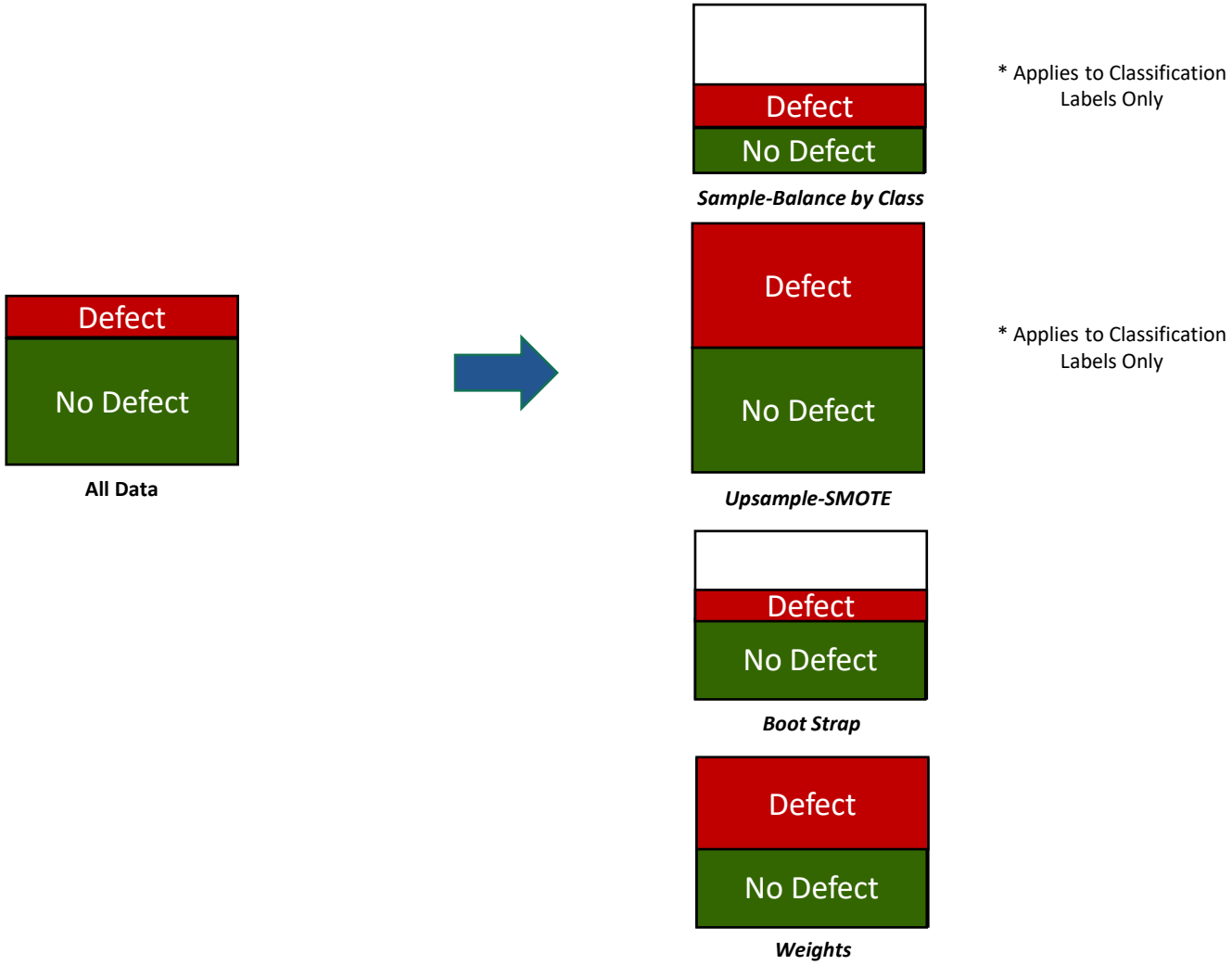
- Recommended Pre-Processing
- Recipe Steps

Table A.1: Preprocessing methods for different models.

model	dummy	zv	impute	decorrelate	normalize	transform
C5_rules()	×	×	×	×	×	×
bag_mars()	✓	×	✓	○	×	○
bag_tree()	×	×	×	○[1]	×	×
bart()	×	×	×	○[1]	×	×
boost_tree()	×[2]	○	✓[2]	○[1]	×	×
cubist_rules()	×	×	×	×	×	×
decision_tree()	×	×	×	○[1]	×	×
discrim_flexible()	✓	×	✓	✓	×	○
discrim_linear()	✓	✓	✓	✓	×	○
discrim_regularized()	✓	✓	✓	✓	×	○
gen_additive_mod()	✓	✓	✓	✓	×	○
linear_reg()	✓	✓	✓	✓	×	○
logistic_reg()	✓	✓	✓	✓	×	○
mars()	✓	×	✓	○	×	○
mlp()	✓	✓	✓	✓	✓	✓
multinom_reg()	✓	✓	✓	✓	×[2]	○
naive_Bayes()	×	✓	✓	○[1]	×	×
nearest_neighbor()	✓	✓	✓	○	✓	✓
pls()	✓	✓	✓	×	✓	✓

Classification – Unbalanced Data

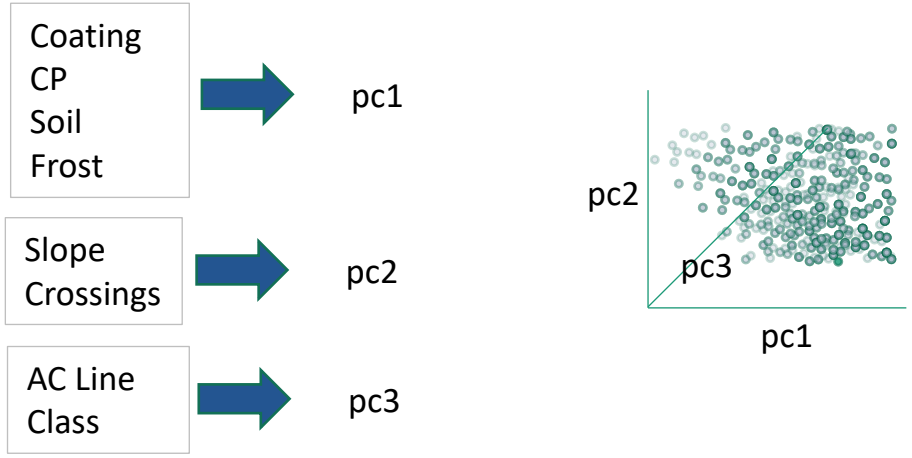
Sampling Strategies – Observations



Feature Engineering

- **Feature engineering** includes methods to transform multiple features into single features to improve the machine learned model:

- **PCA (Principal Component Analysis)** – Principal Component Analysis is a mathematical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated attributes into a set of values of uncorrelated attributes called principal components (think GPA):



Feature Selection

- **Feature selection** includes methods to select features based on their ability to improve the ***Performance (Minimize Error)*** of the machine learned model:

Backward Elimination

Accuracy	Coat	CP	Soil Ph	Frost	X-Ings
77%	1	2	3	4	5
80%	1	2	3		4
93%	1	2			3
89%			Stop		

- CP_On
- CP_Off
- Ext_Coating
- Power_Line
- Diameter
- Install_Yr
- Seam
- Frost
- RAILROAD
- Corrosivity
- Structures
- DOC
- Water_Body
- Farmland
- Bedrock
- Flooding
- Slope
- Stream
- Hwy_Type
- Road_Type

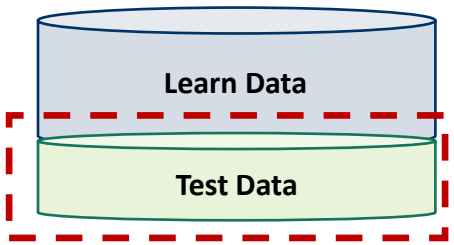
Forward Selection

Accuracy	Coat	CP	Soil Ph	Frost	X-Ings
84%		1			
87%	2	1			
93%	2	1			3
89%			Stop		

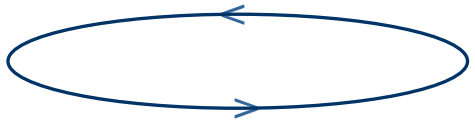
MODEL VALIDATION & TUNING

Machine Learning Process - Refresher

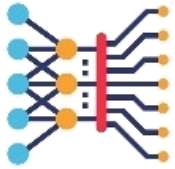
Training Data
(Observations)



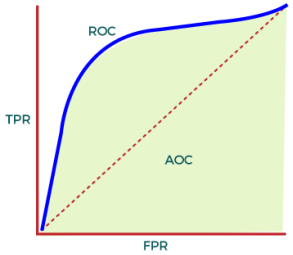
Target of Interest
(Categorical\Numerical)



Learned Model
(Target)



Model Performance
(Validation & Acceptance)



Model Validation & Tuning

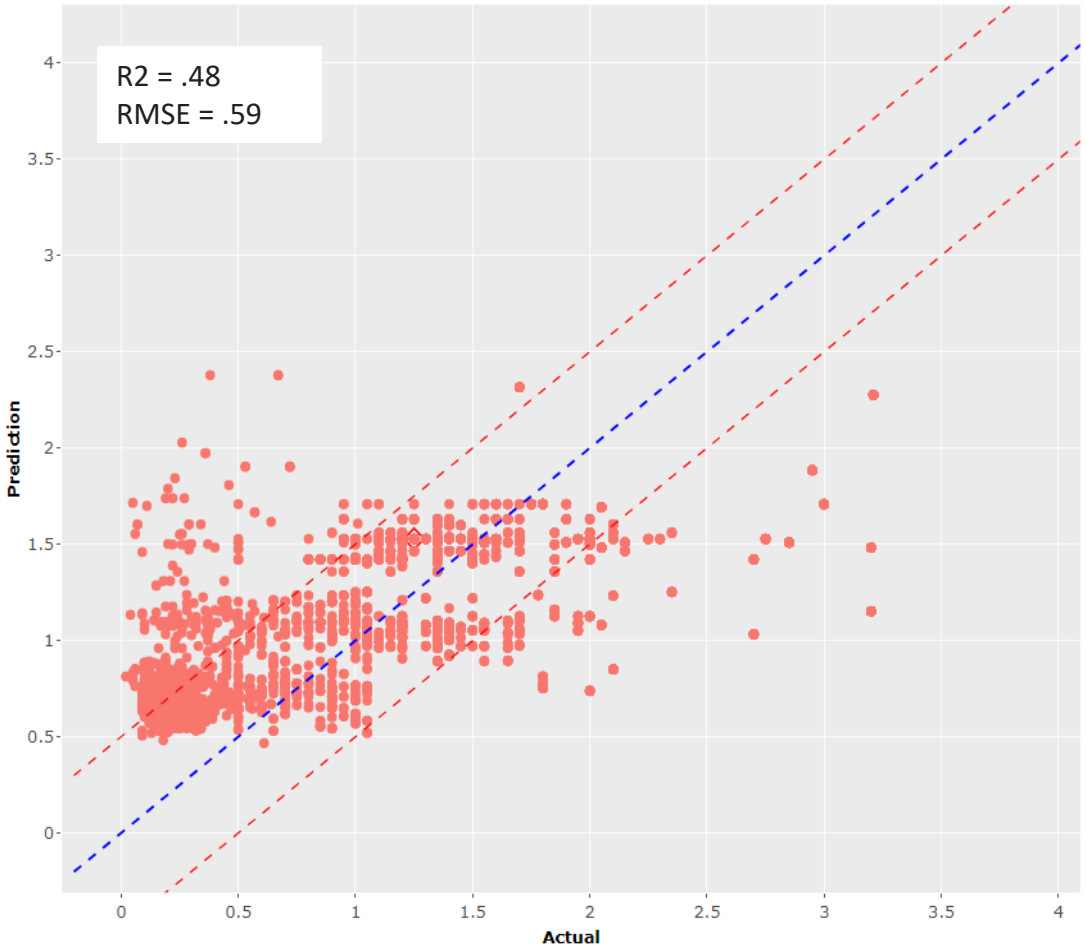
Unit 3.1

Model Error

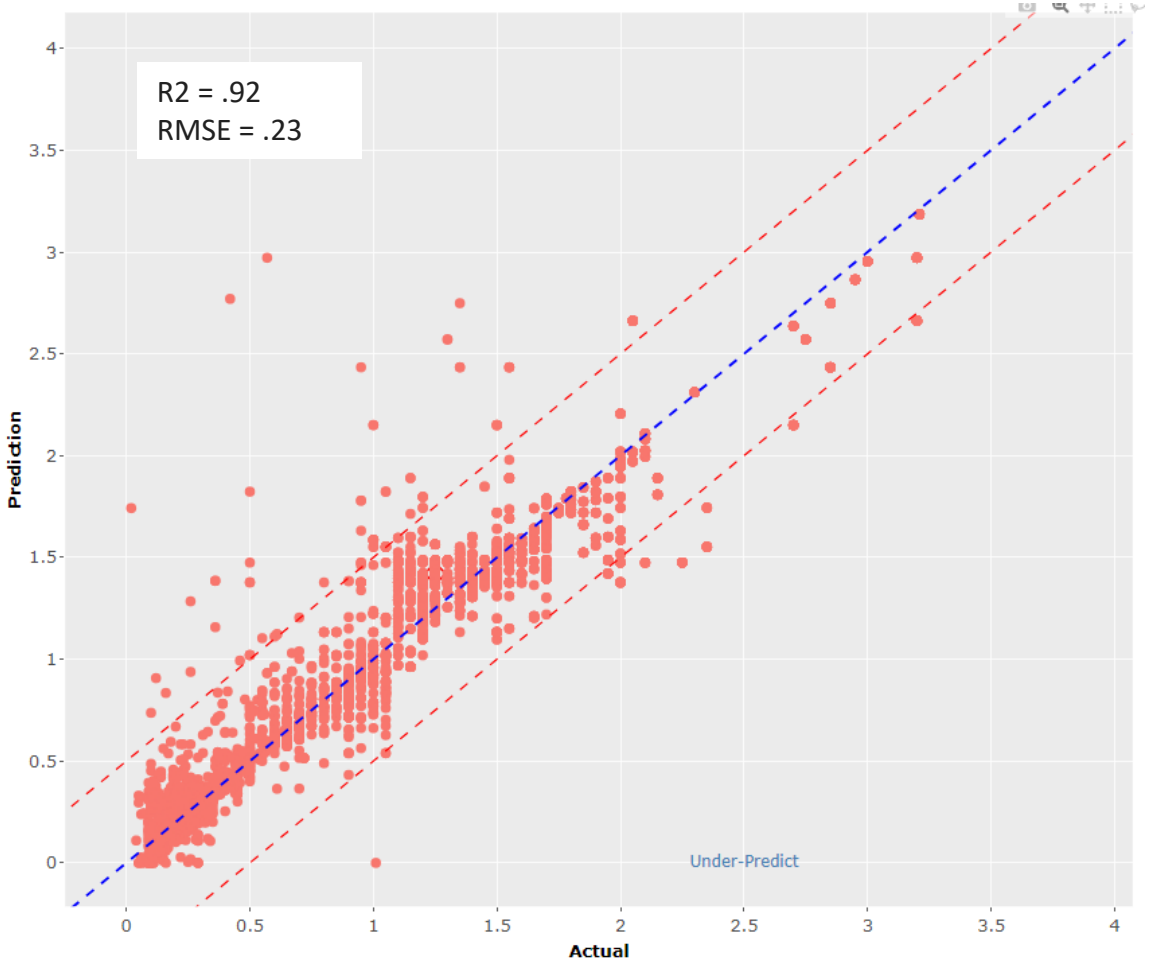
Regression Model Error

Regression Model Performance – Test with Unseen Data

Un-Tuned Model



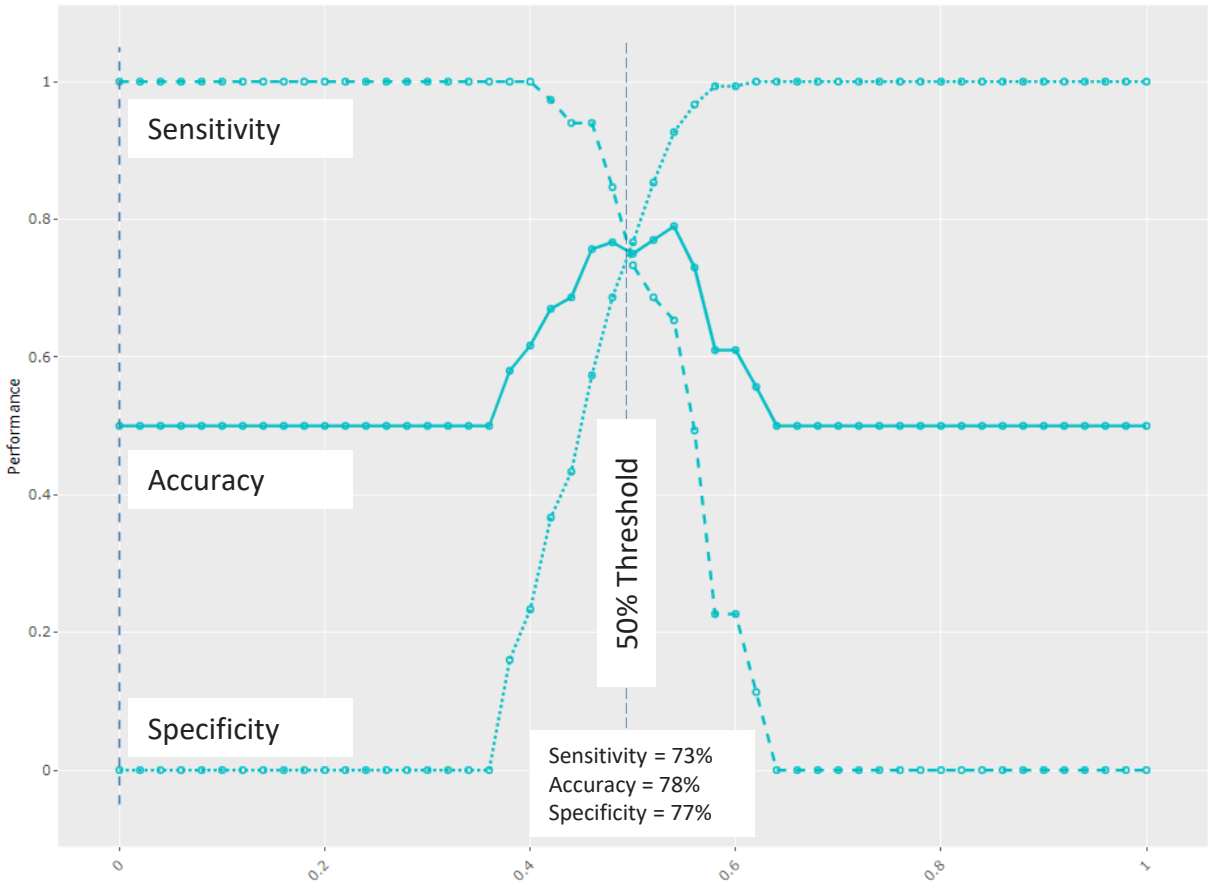
Tuned Model



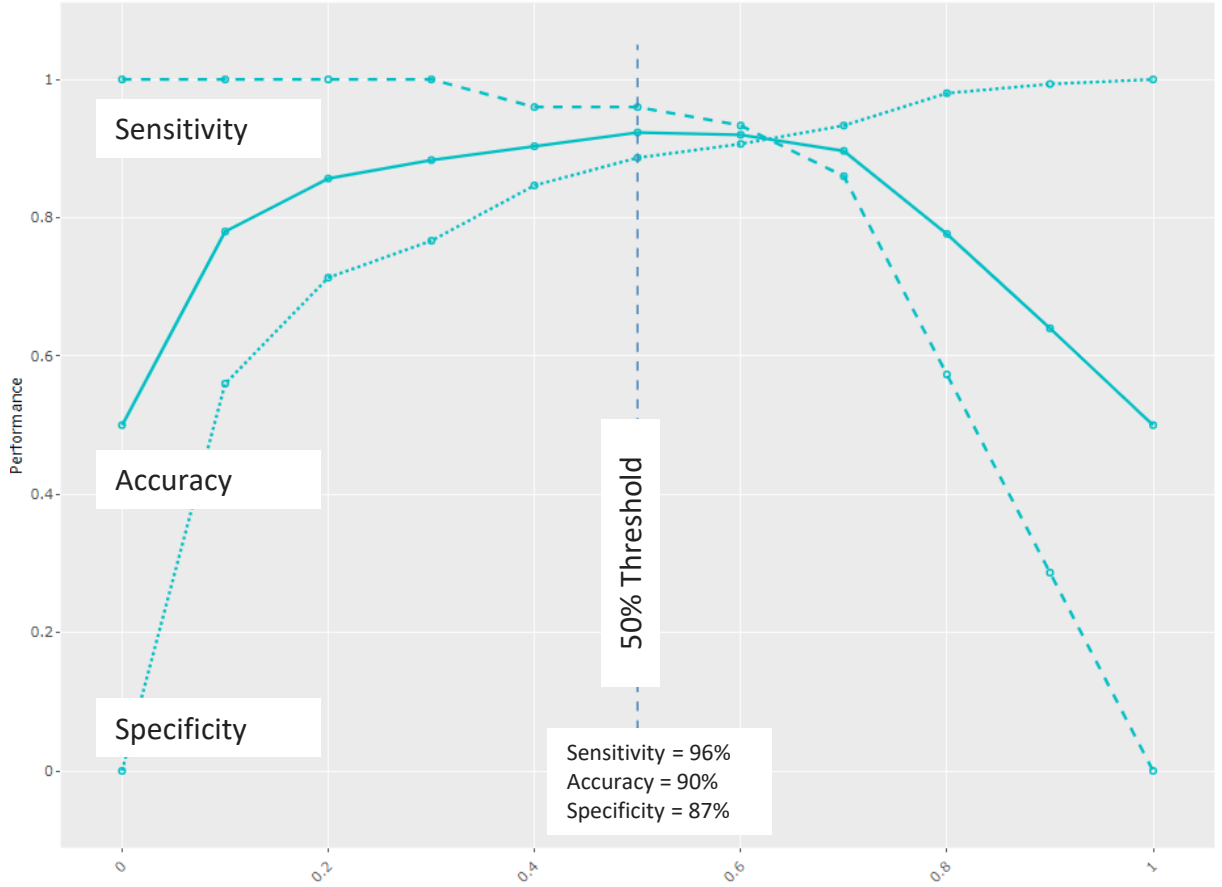
Classification Model Error

Classification Model Performance – Test with Unseen Data

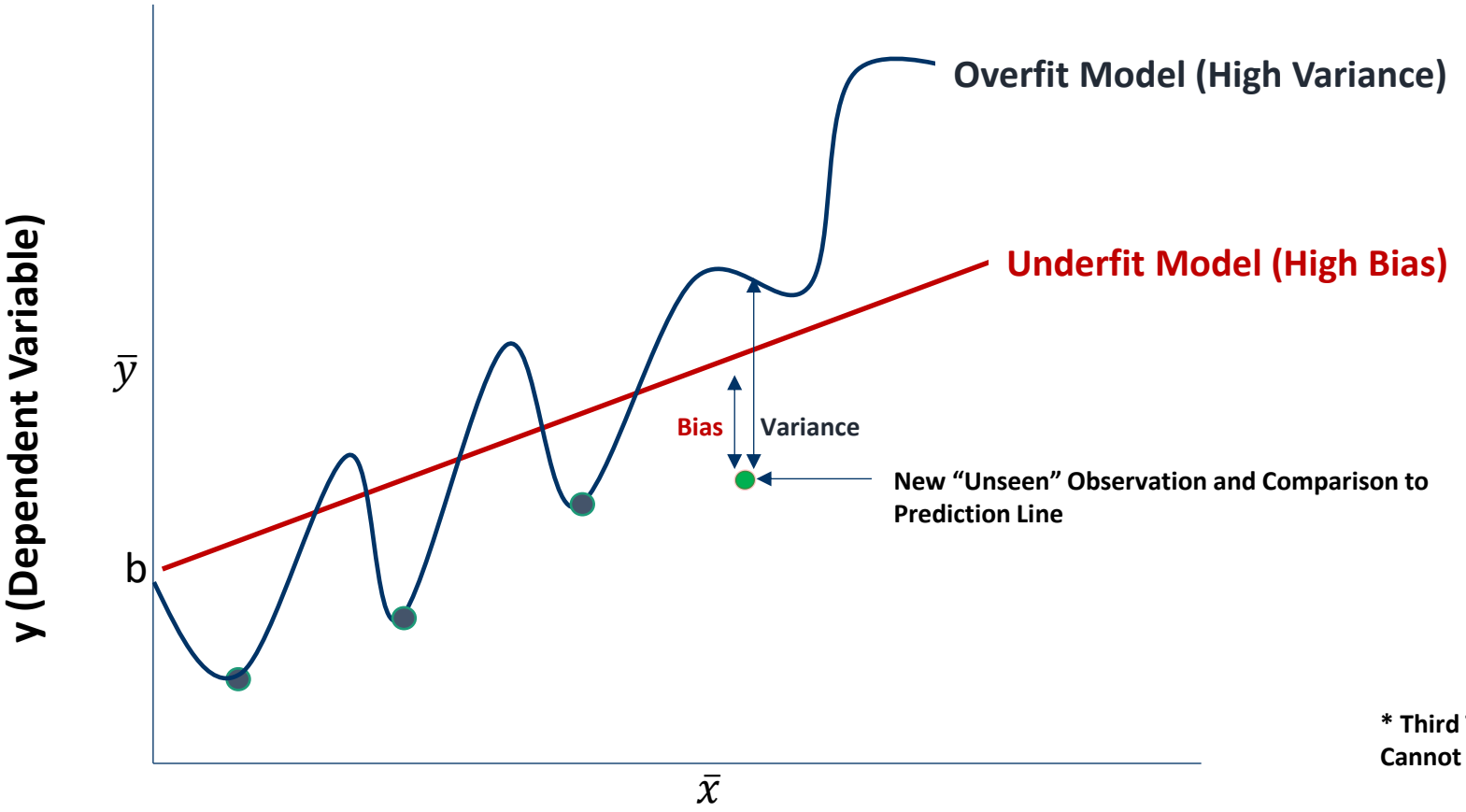
Un-Tuned Model



Tuned Model



Error Types - Bias & Variance*



* Third Type of Error is "Irreducible" Error which Cannot be Reduced

Model Validation & Tuning

Unit 3.2

Model Tuning

Model Tuning – Best Practices for Learning the “Best” Model

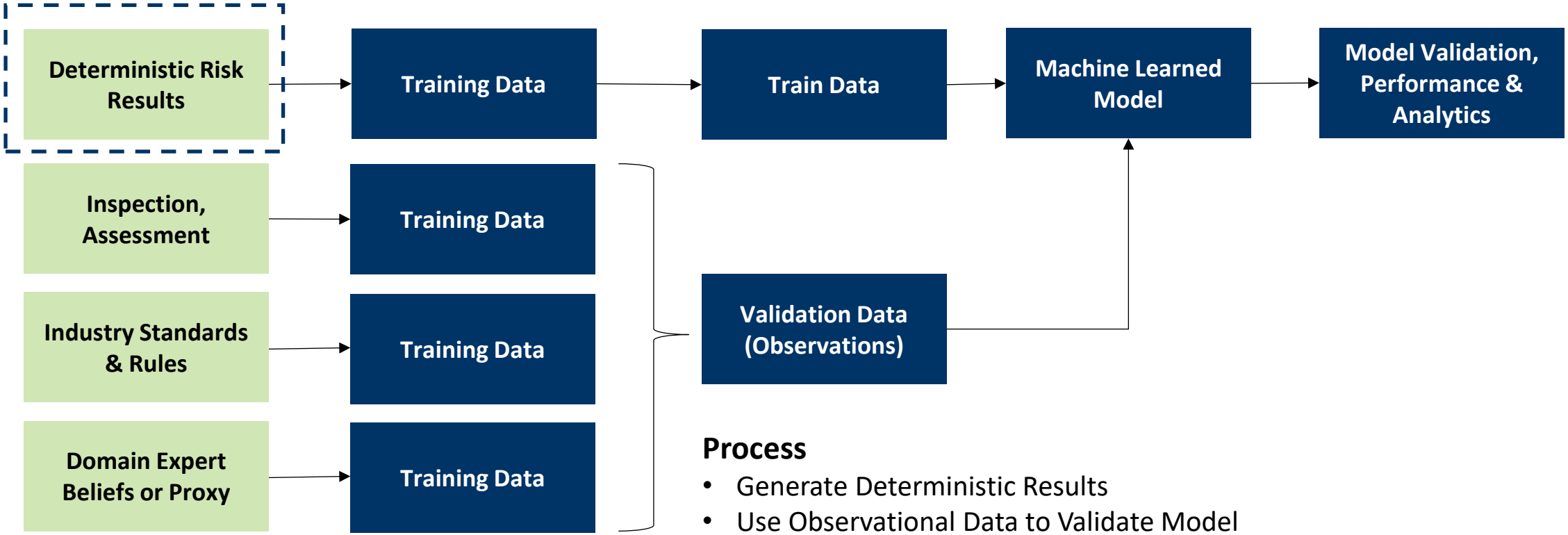
- ❑ Domain Expert Review of Predictors (Modify, Add, Remove Predictors)
- ❑ Try Different Learning Methods
- ❑ Optimize Method Parameters (Use Hyper-Parameter Grid Search)
- ❑ Add, Remove Predictors thru Forward Selection
- ❑ Add, Remove Predictors based on Global & Local Sensitivity Analysis Methods
- ❑ Review & Improve Learning Observation Assumptions
- ❑ Review & Improve Learning Data Sampling (Use Sampling Learning Curve)
- ❑ Use Weights to Increase\Decrease Importance of Learning Observations
- ❑ Explore Pre-Processing Options (Encoding, Up\Down Sampling, Correlations, etc.)
- ❑ Explore Feature Engineering (Combine or Modify Features, PCA)
- ❑ Investigate Incorrect Predictions (Sensitivity Analysis)
- ❑ Use Model Simulation with Domain Experts

Model Validation & Tuning

Unit 3.3

Deterministic Model Validation

Validating a Deterministic Model



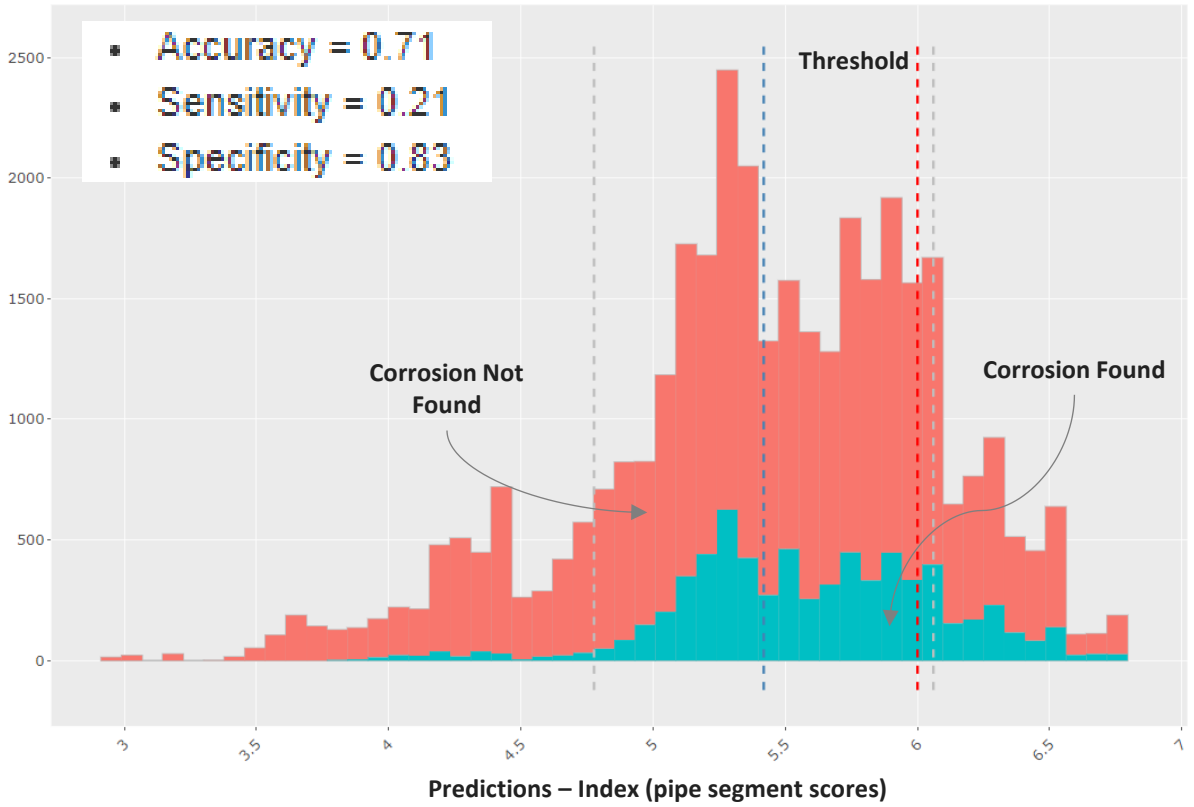
Process

- Generate Deterministic Results
- Use Observational Data to Validate Model
  - Use Binary Positive\Negative Observation (T/F, Yes/No, etc.), or
  - Use Measured Value in Same Units as Deterministic Results
- Modify Deterministic Structure to Improve Model, or
- Learn a New Model with Same Predictors but based on Observations

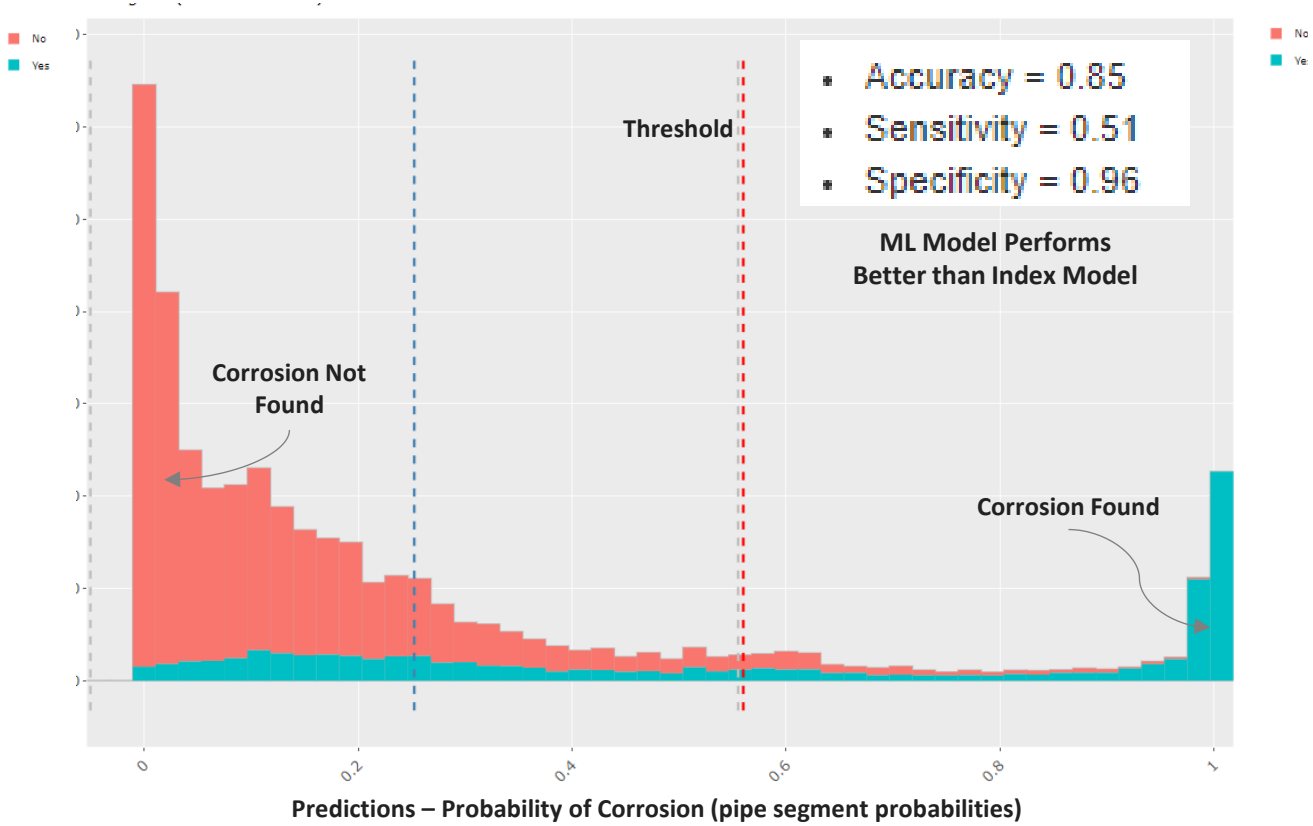
Deterministic vs. Machine Learned Performance

External Corrosion Example (Same Predictor Data - Deterministic vs. Machine Learned both Tested w\Observations)

Deterministic Model



Machine Learned Model



Model Learned based on Deterministic Structure

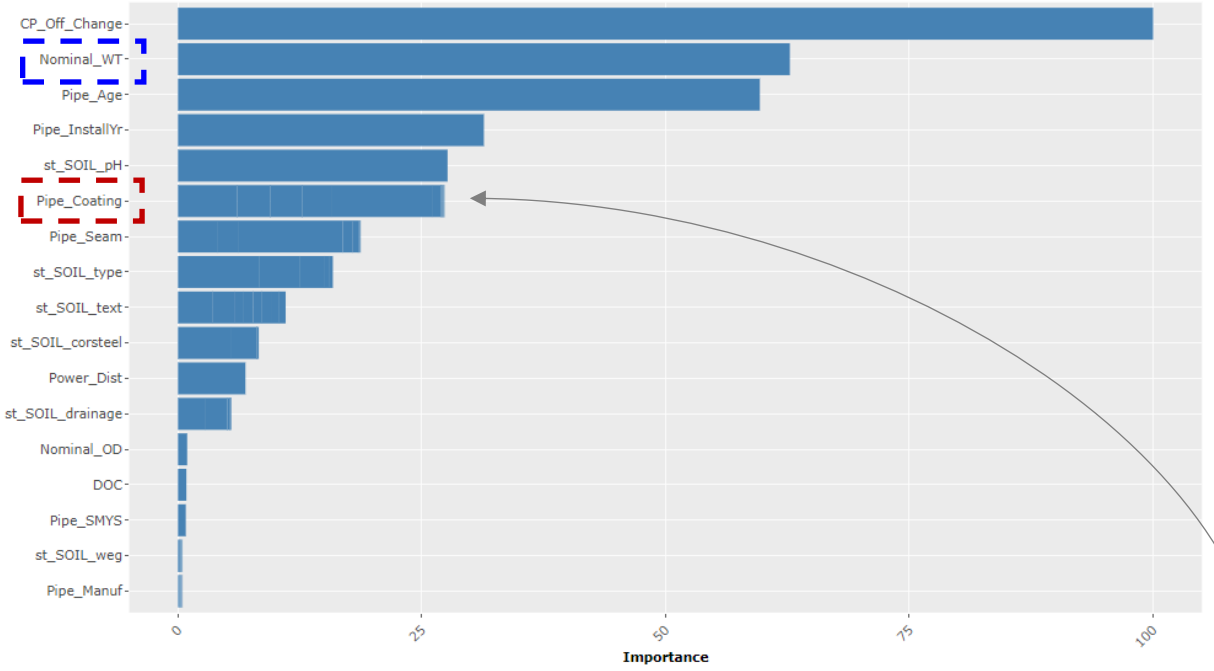
Model Learned with Observational Data

*** Performance Based on Threshold = One SD from Mean**

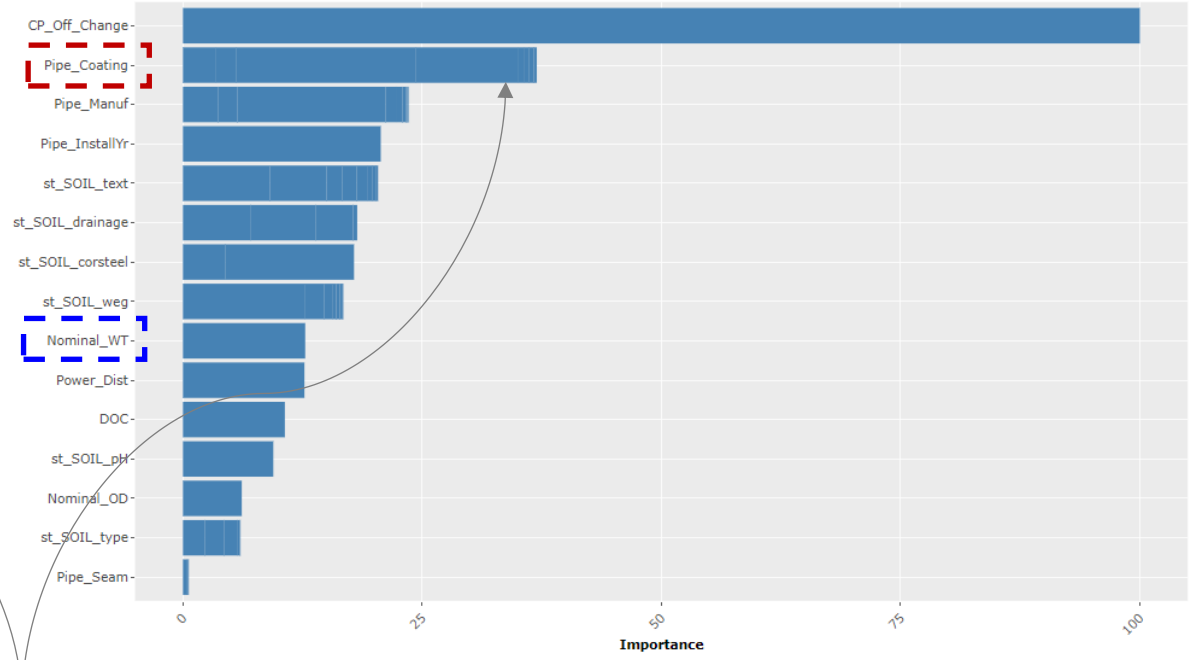
Model Insights – Predictor Influence (Global)

Predictor Importance Normalized Weights

Deterministic Model



Machine Learned Model

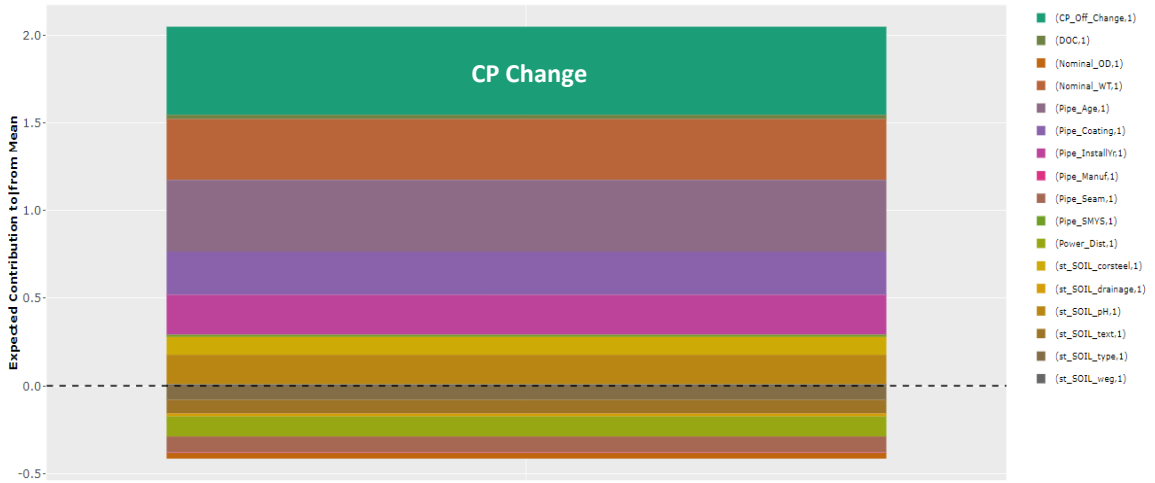


Deterministic vs. Machine Learned Weights Vary Between Approaches

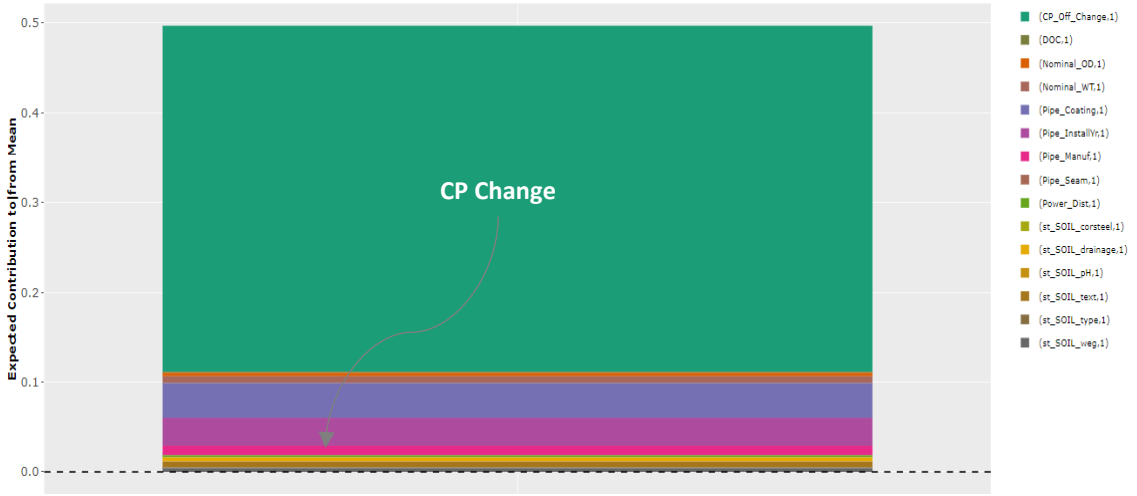
Model Insights – Predictor Influence (Local)

Pipe Segment Predictor Contributions (Top 100 Aggregated Predictions)

Deterministic Model



Machine Learned Model



Predictor Contributions Vary Between Approaches (Machine Learning Considers Predictor Non-Linearities and Interactions)

Machine Learned Based Risk

Unit 4.1

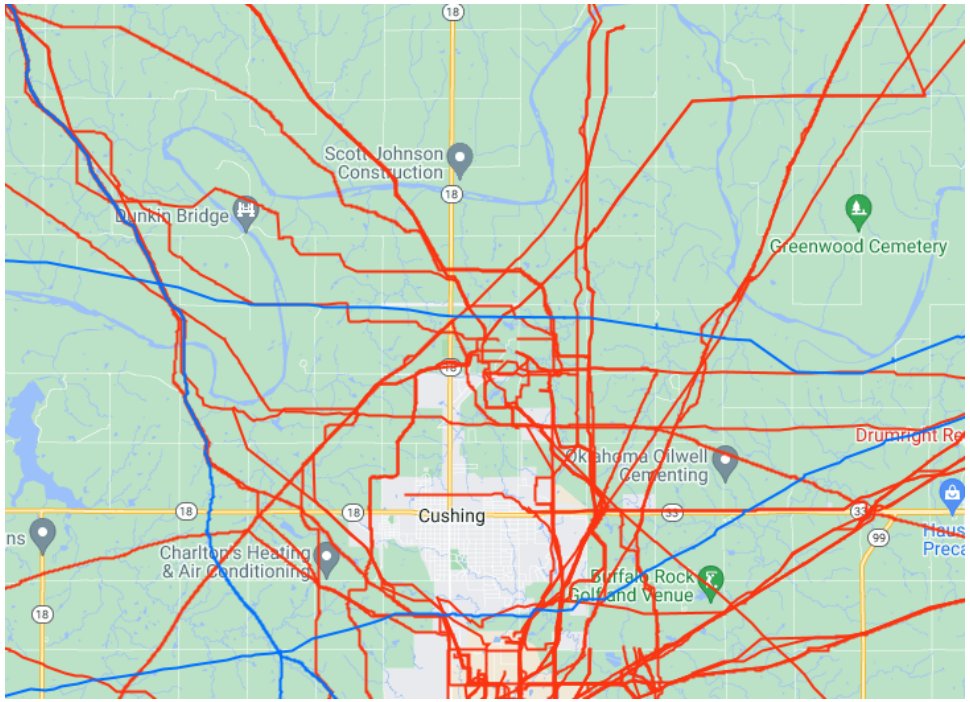
Overview

Risk Management

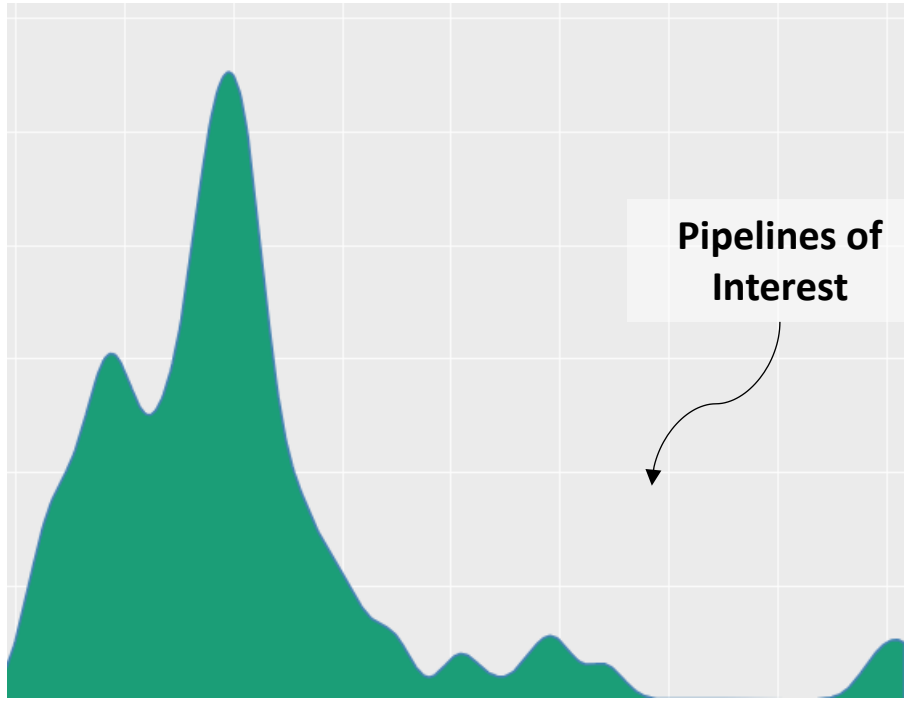
Purpose Risk Process

- Assess Risk & Prioritize Assets
- Identify Best Inspection & Mitigation Options
- Meet Compliance Requirements

Pipeline System



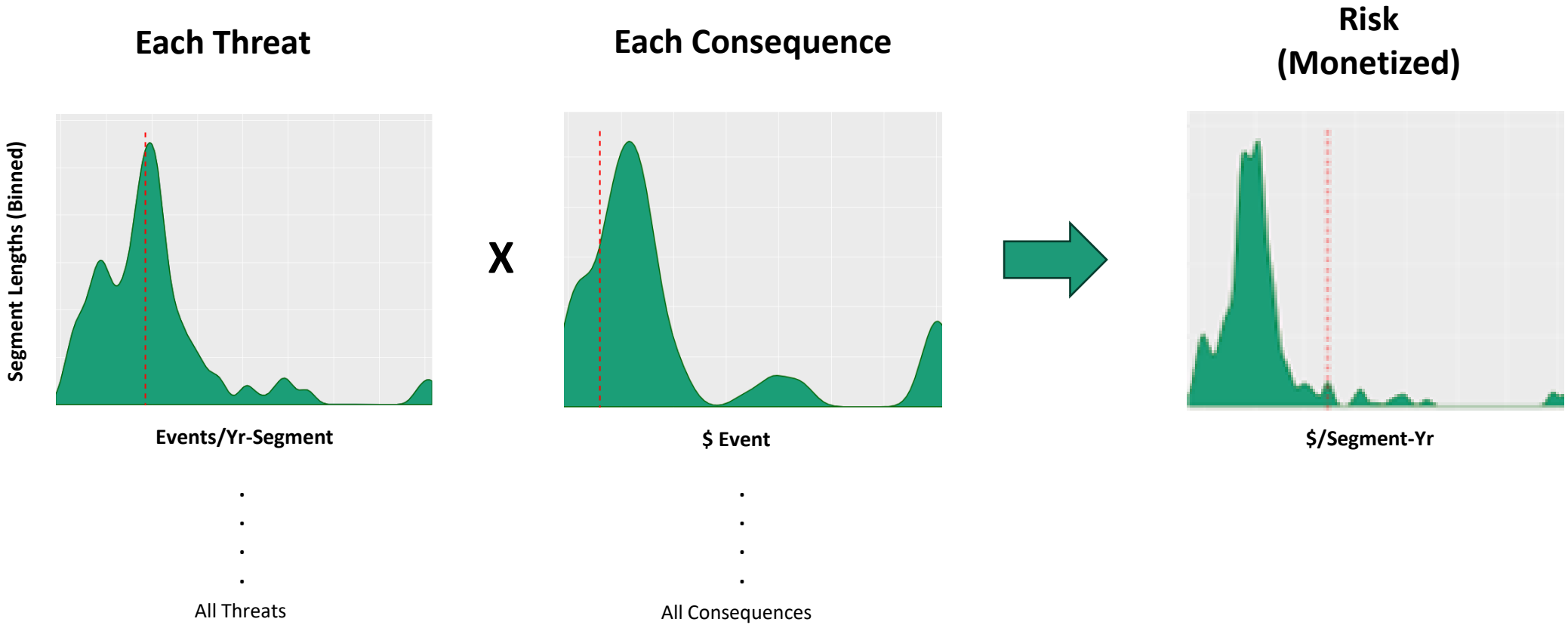
Risk Profile



Risk Management

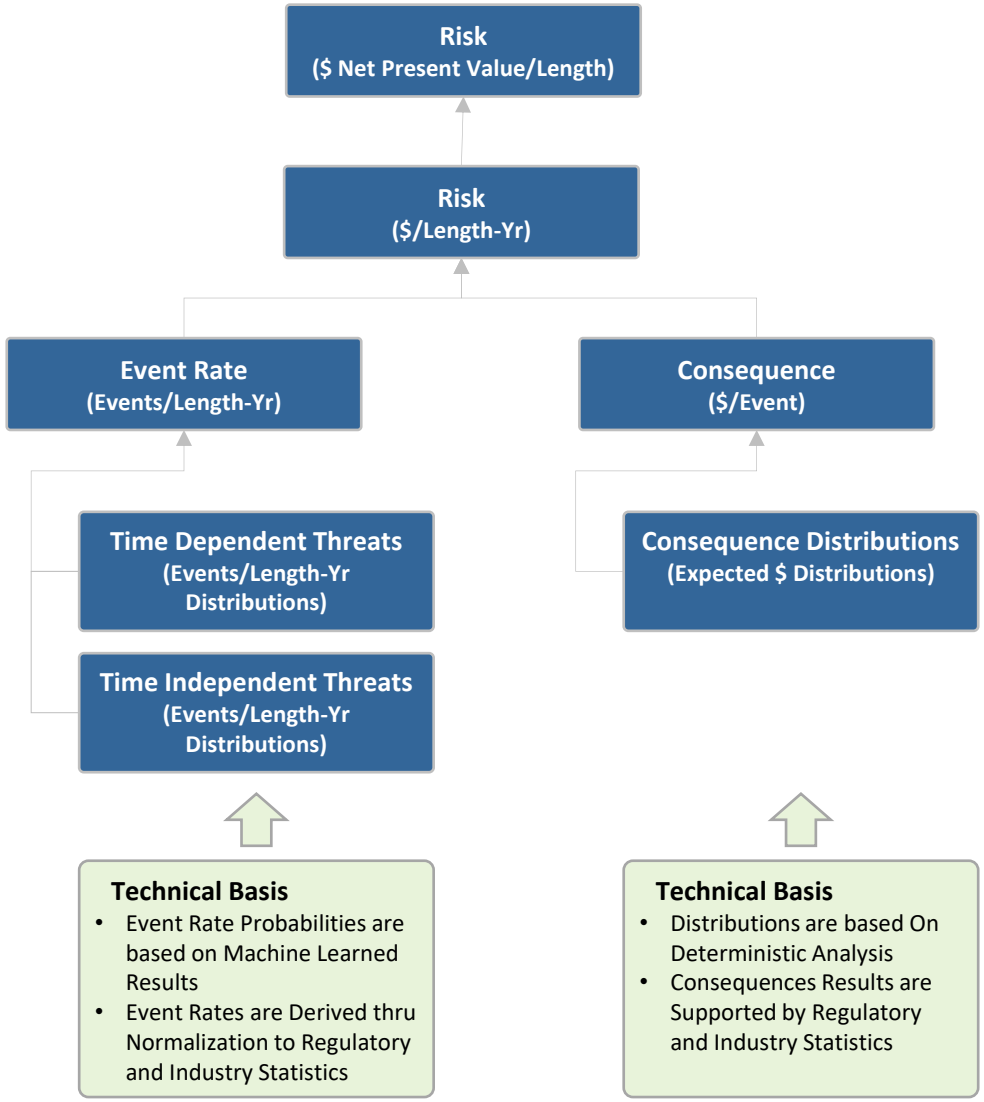
Benefits of Machine Learned Models

- Data Driven based on Observations & Beliefs
- Explicitly Validated & Tuned, Transparent & Explainable
- Considers Data Non-Linearities & Interactions
- Supports Probabilistic and Quantitative Analysis



A Structure for Machine Learning Based Risk Management

Structure

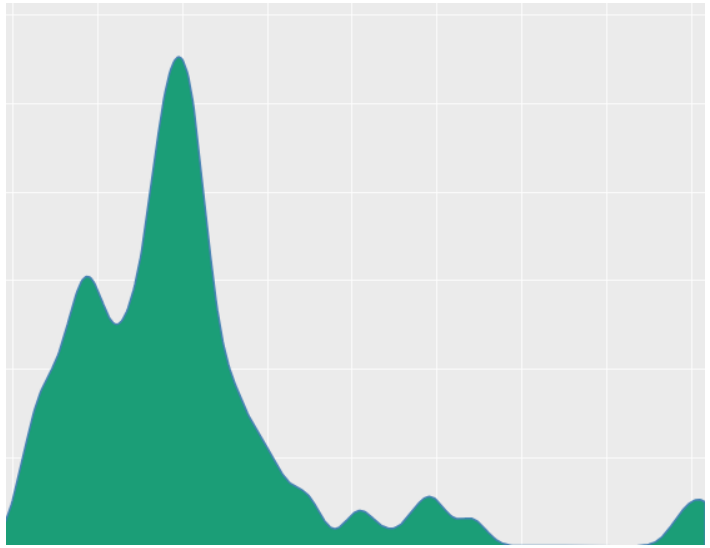


Key Points

- Monetized Risk Supports Financial Planning, Mitigation Decision-Making & Compliance Requirements
- Analysis may be Performed for Pipelines ($/Length) or Fixed Assets ($/Asset)
- Risk is based on a Machine Learned Estimate of Unwanted Future Events times an Estimate of Potential Consequences of the Event
- Events are Categorized as Either Time Dependent or Time Independent
- Time Dependent Events can Change Over Time and are based on ML Based Survival Curves with Probability of Events Over Time
- Time Independent Events are not Expected to Change Over Time and are Based on ML Based Probabilities
- Ensemble Models (Combinations of ML Models) may be Created to Improve Prediction Efficacy
- Risk Roll-Up is Possible as the Common Unit of Measurement is $ Net Present Value of Carried Risk for all Threats by Length or Asset
- The Practitioner Leverages ML Results at Different Levels Depending on Requirements (i.e. $/mile, $/segment, max. POER in segment or per mile, etc.)

Normalizing Distributions to Expected Values

Probability of Event Rate Distribution



min

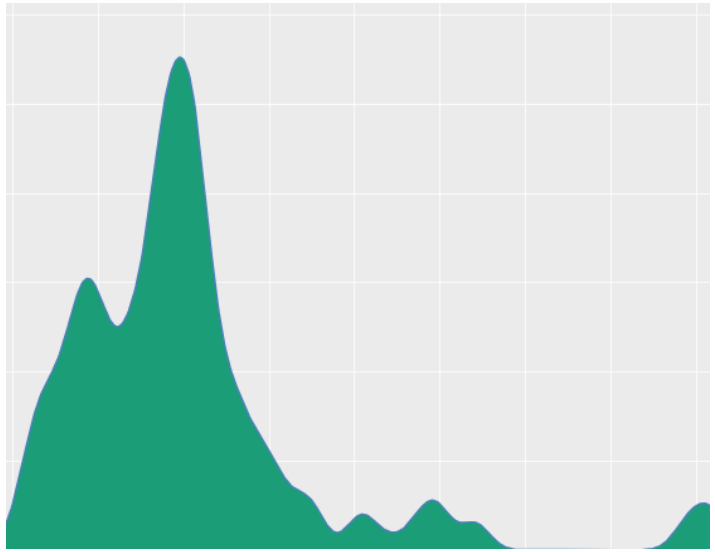
max

Machine Learned Output
(Probability % of Event Rate)

Scaling Function based on Assumed Event Rate

- Company Incident History
- PHMSA Industry History

Event Rate Distribution

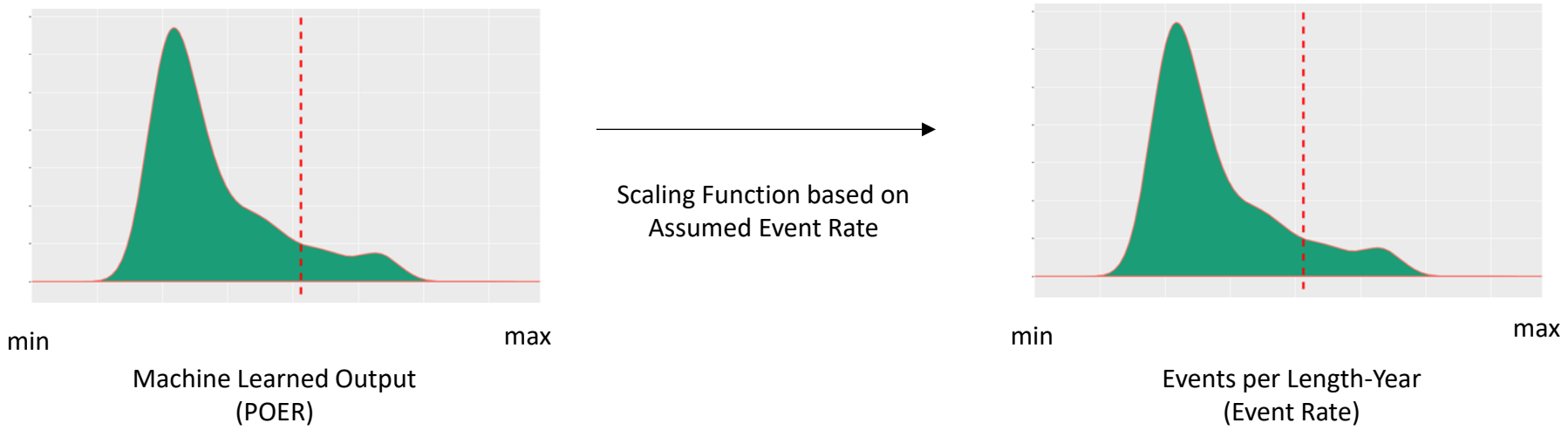


min

max

Events per Length-Year
(Event Rate)

Machine Learned Event Rates to Risk



Key Points

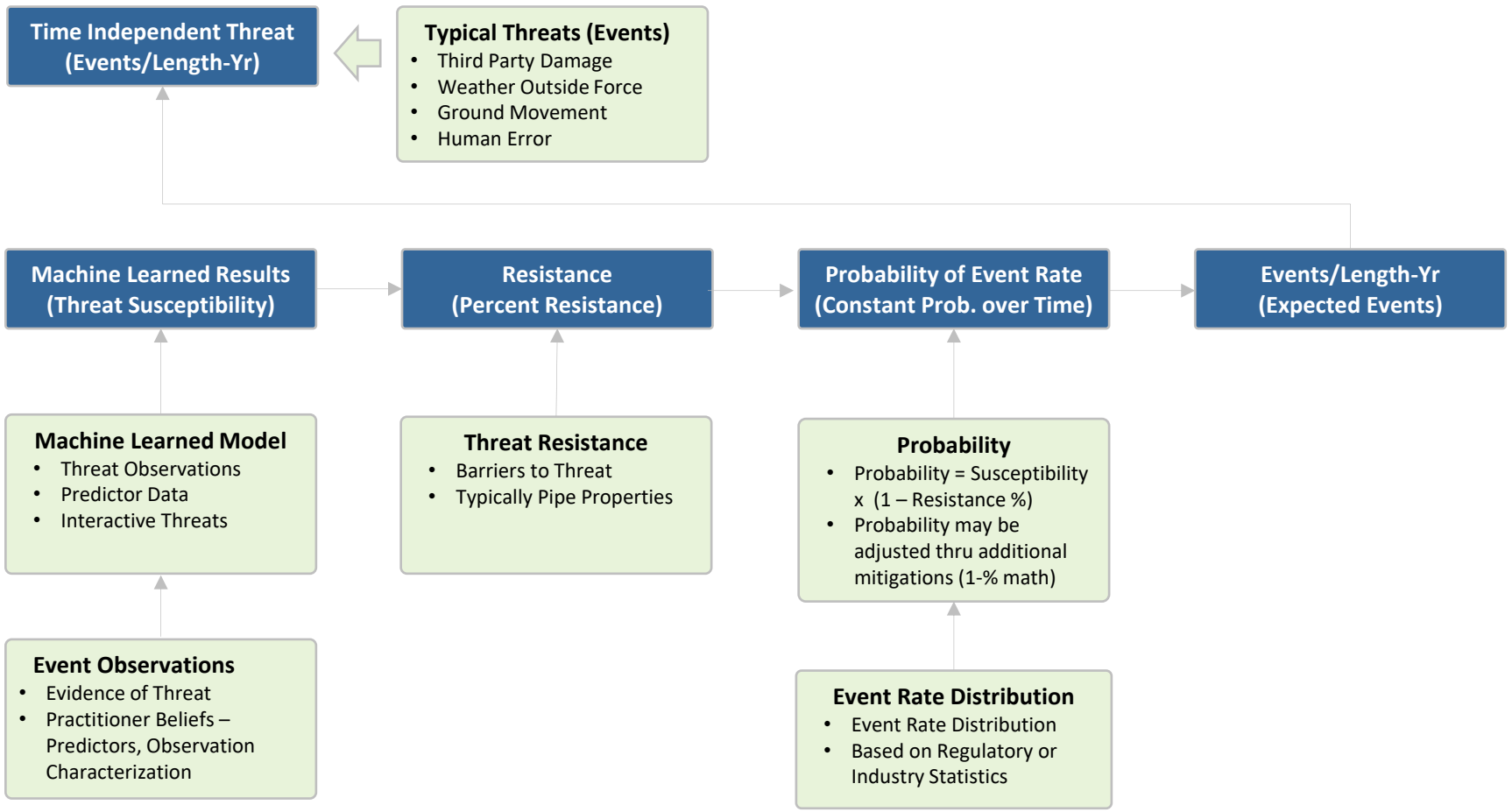
- **POER (Time Independent)** - The output of a machine learned time independent threat model is typically a probability of true or false. We call this a data driven probability of an event rate (POER) which may be further adjusted by % mitigation and\or % resistance

- **POER (Time Dependent) -** The output of a machine learned time dependent threat model is typically a corrosion or growth rate which we use to calculate a time to criteria (TTC). We use Weibull equations to convert this to a probability (based on learned shape and time to failure parameters) and call this a data driven probability of an event rate (POER) which may be further adjusted by % mitigation and\or % resistance

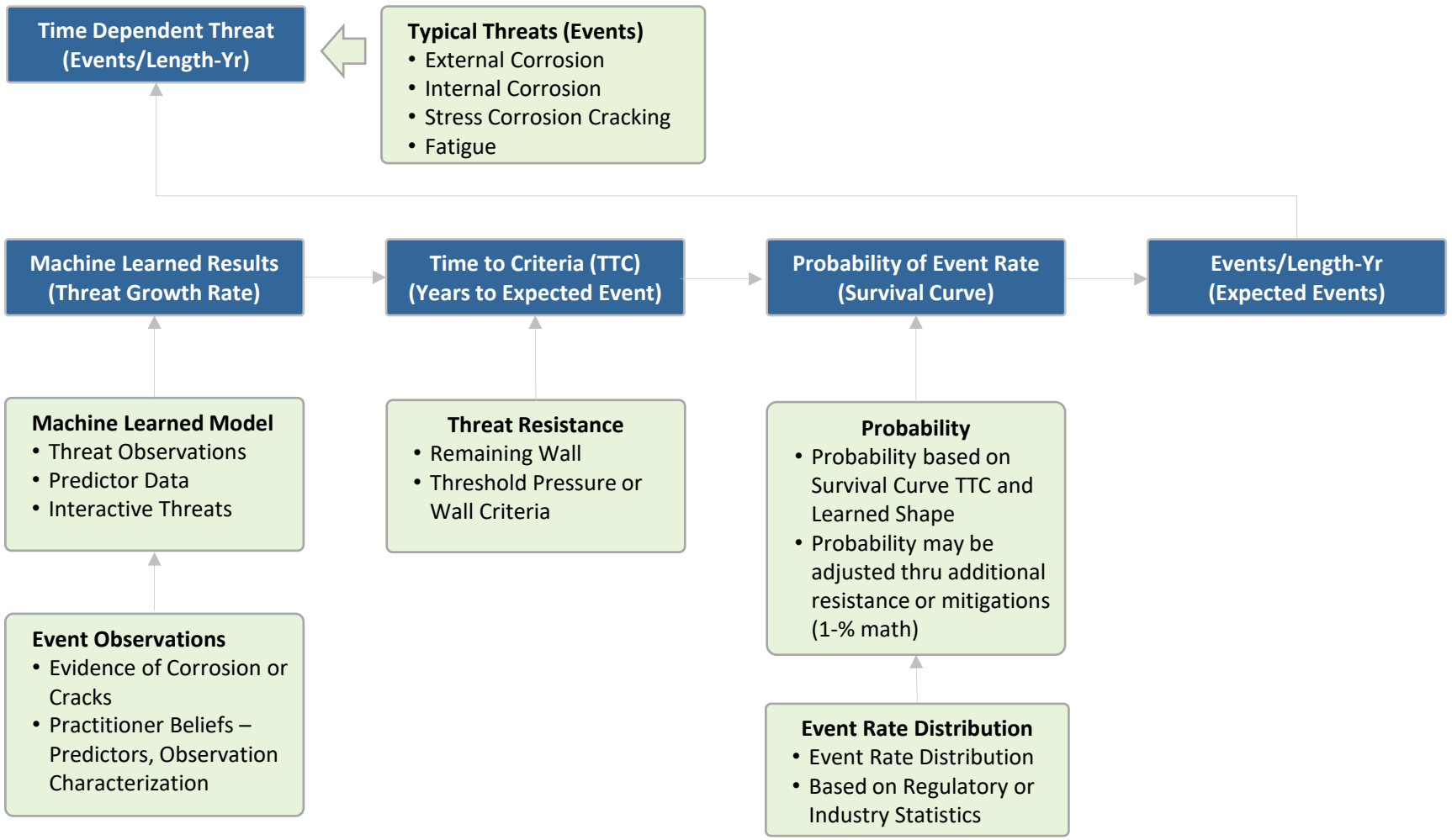
- **Event Rates -** POER distributions are data driven and are the basis of output event rates. Event rates are required to get to a quantitative output such as expected events per year and monetized risk. Without this normalization near-real world interpretation and application of output results are limited for both deterministic and machine learning structures.

- **Risk -** The process is to scale POER distributions to an event rate (i.e., unwanted events/length-year) based on regulatory, industry or asset owner histories or expectations, like actuary tables used by other industries. Event rates are then easily converted to expected events in a year for a pipe segment and may be multiplied by expected consequences to quantify risk for that pipe segment for a given year.

Time Independent Threats



Time Dependent Threats

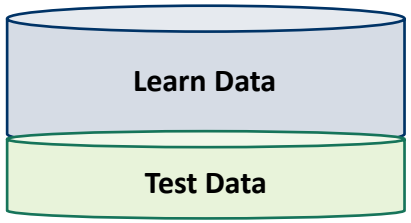


Course Closing

USE CASES

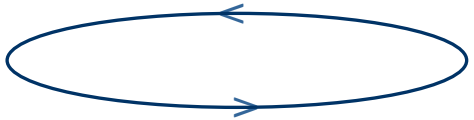
Machine Learning Process

Training Data
(Observations)



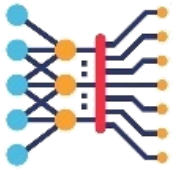
Training Data:
- Inspection Results
- Domain Expertise
- Incidents & Leaks
- Deterministic Results

Target of Interest
(Categorical\Numerical)



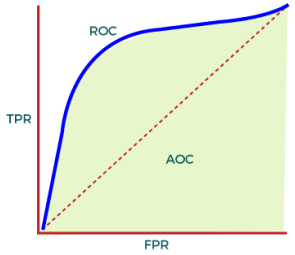
Learning Targets:
- Threats, Consequences
- Data Quality, Missing Data
- Program Costs
- Time to Criteria

Learned Model
(Target)



Models:
- Pattern Recognition
- Many Methods
- Model Tuning
- Predictive vs. Inference

Performance & Insights
(Validation & Acceptance)



Performance:
- Learning Curves & Unity Plots
- Uncertainty
- Model Acceptance
- Predictor Influence, Directionality & Contribution

Technical Notes

Typical ML Processes
- Supervised (shown above)
- Unsupervised (no observations)
- Semi-Supervised
- Self-Supervised
- Synthetic Data Learning

Typical Targets
- Numerical (Regression)
- Two-Class (Classification)
- Multi-Class (Multi-Classification)

Models
- Hundreds of Methods
- Predictive
- Inferential (Explanatory)
- Ensembles

Typical Performance Metrics
- ROC, AUC, Accuracy
- Sensitivity, Specificity
- R2, RMSE, MAE
- KAPA, F1

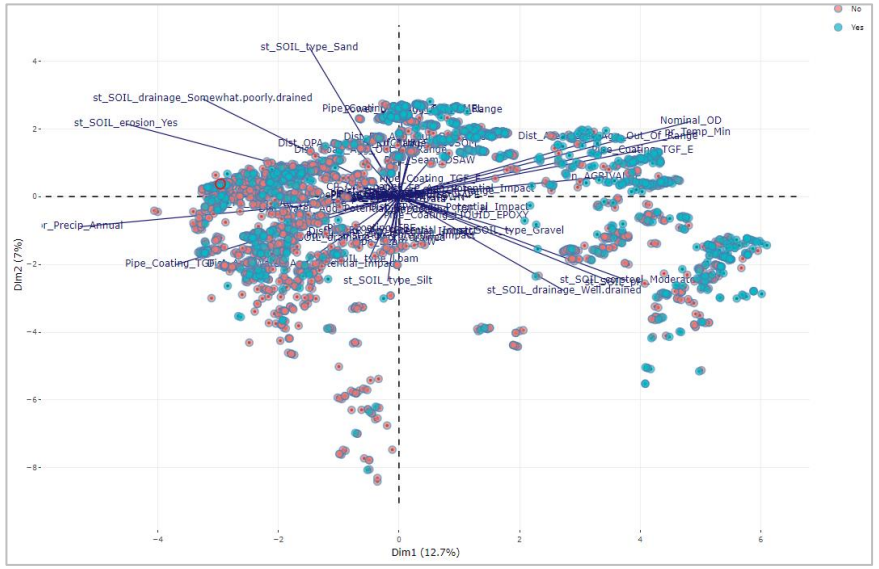
USE CASES
Supporting Documentation

Assess Model Applicability

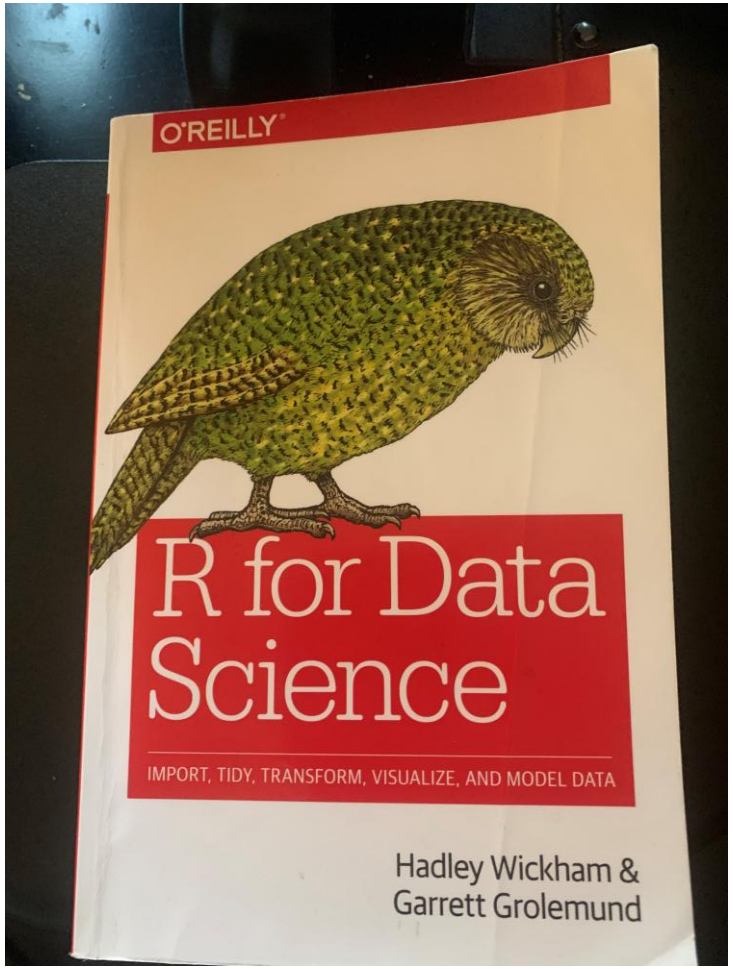
Assess Model Applicability

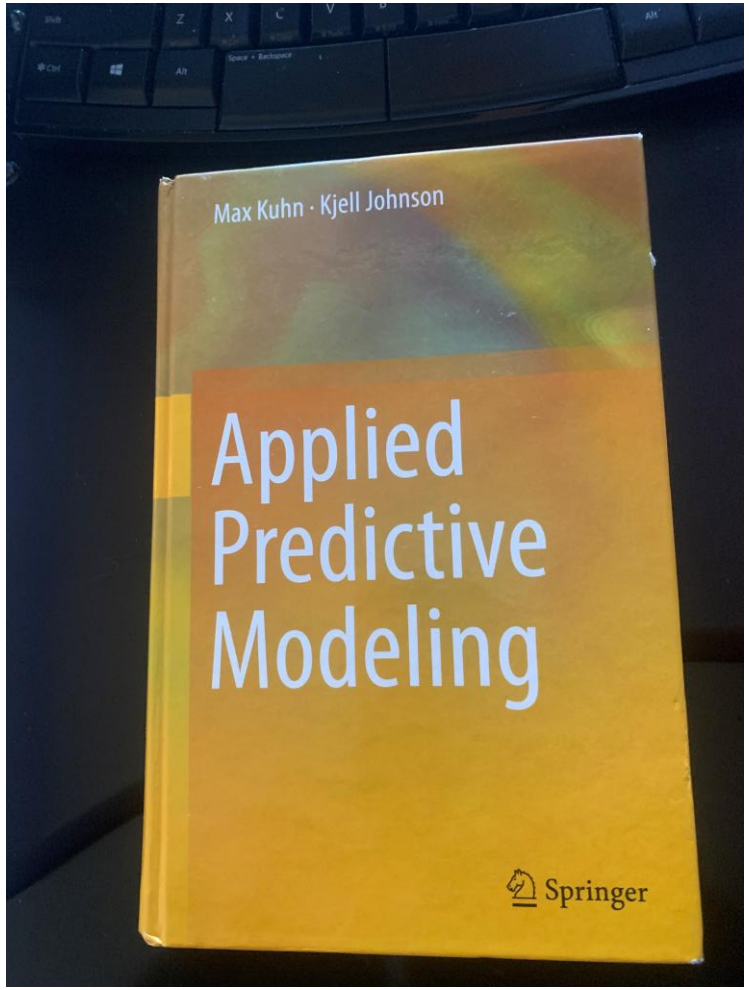
How can we Test if a Learned Model can be used for New Prediction Data?

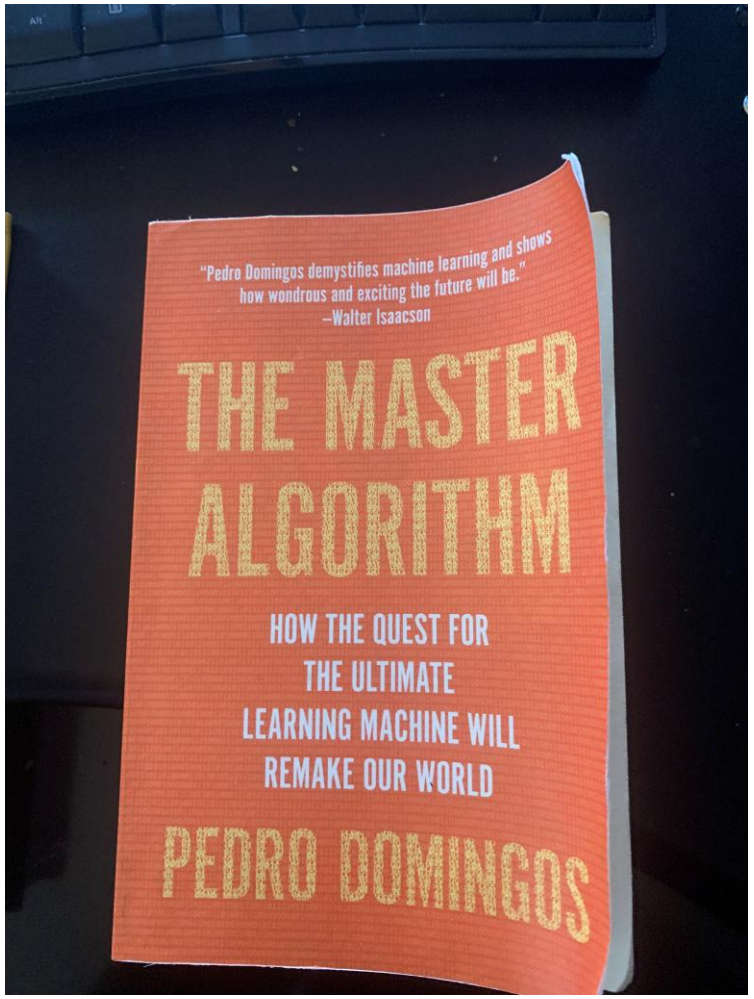
- Statistical Comparison (Variance Report)
- t-SNE Analysis
- PCA Analysis
- Learned Applicability Model Analysis
- Compare Prediction Distributions (Learn vs. Predict)

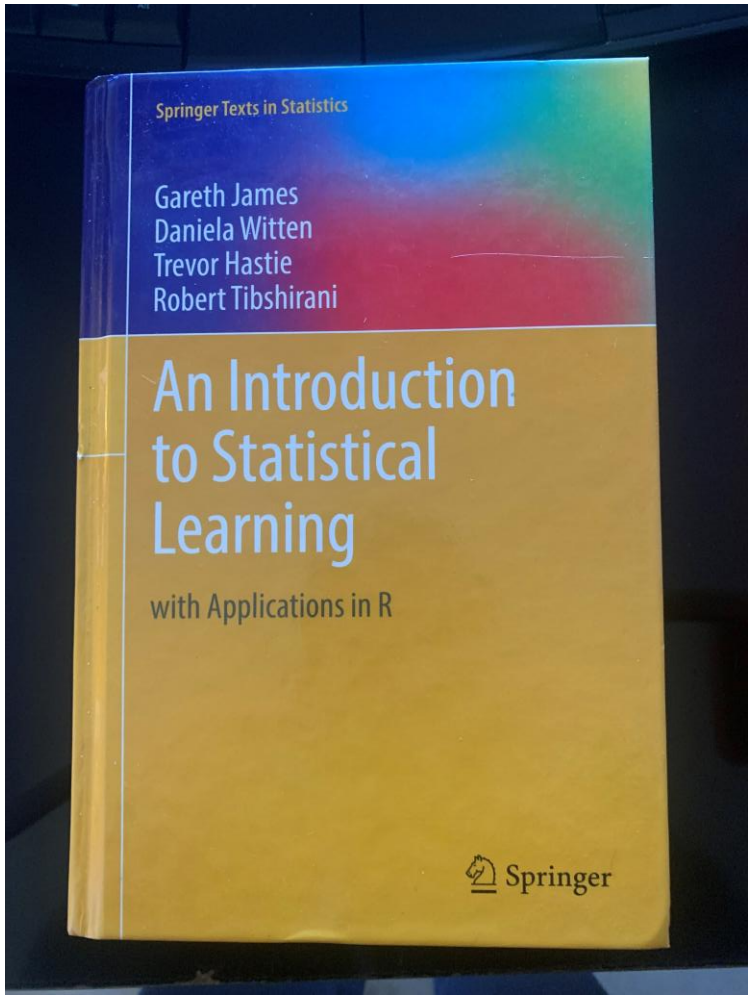


Resources









Extra Slides

Applicability Model

Target

Learning Data

Rec_ID	Level1	Level2	Comp_Name	Comp_ID	EC_BINARY	Coating	Comp_Distance	CP_Off_Trend	Crossing_Water	Depth_Cover	Diameter
2713	Level1	Level2	Pipeline_1	1	Learn	TGF_E	229,504.00	-0.01	None	18.00	30.00
3781	Level1	Level2	Pipeline_1	1		LIQUID_EPOXY	151,167.00	-0.01	None	24.00	30.00
2780	Level1	Level2	Pipeline_1	1		TGF_F	106,414.00	-0.02	None	12.00	30.00
20834	Level1	Level2	Pipeline_5	5		TGF_A	54,290.00	0.02	None	72.00	18.00
9618	Level1	Level2	Pipeline_3	3		TGF_E	24,493.00	-0.02	None	24.00	30.00
6321	Level1	Level2	Pipeline_2	2		TGF_E	80,597.00	-0.02	BRANCH	30.00	30.00
8039	Level1	Level2	Pipeline_2	2		POLY	92,628.00	0.00	None	18.00	30.00
24392	Level1	Level2	Pipeline_6	6		TGF_A	74,903.00	-0.01	None	18.00	18.00
5137	Level1	Level2	Pipeline_2	2		FBE	94,171.00	0.01	None	30.00	30.00
12097	Level1	Level2	Pipeline_3	3		TGF_E	25,975.00	-0.00	None	24.00	30.00
5202	Level1	Level2	Pipeline_2	2		TGF_F	72,565.00	-0.00	None	78.00	30.00
20076	Level1	Level2	Pipeline_5	5		TGF_A	29,353.00	0.01	None	64.00	18.00

Random Data

Rec_ID	Level1	Level2	Comp_Name	Comp_ID	EC_BINARY	Coating	Comp_Distance	CP_Off_Trend	Crossing_Water	Depth_Cover	Diameter
2713	Level1	Level2	Pipeline_1	1	Random	TGF_E	229,504.00	-0.01	None	18.00	30.00
3781	Level1	Level2	Pipeline_1	1		LIQUID_EPOXY	151,167.00	-0.01	None	24.00	30.00
2780	Level1	Level2	Pipeline_1	1		TGF_F	106,414.00	-0.02	None	12.00	30.00
20834	Level1	Level2	Pipeline_5	5		TGF_A	54,290.00			72.00	18.00
9618	Level1	Level2	Pipeline_3	3		TGF_E	24,493.00			24.00	30.00
6321	Level1	Level2	Pipeline_2	2		TGF_E	80,597.00			30.00	30.00
8039	Level1	Level2	Pipeline_2	2		POLY	92,628.00			18.00	30.00
24392	Level1	Level2	Pipeline_6	6		TGF_A	74,903.00			18.00	18.00
5137	Level1	Level2	Pipeline_2	2		FBE	94,171.00	0.01	None	30.00	30.00
12097	Level1	Level2	Pipeline_3	3		TGF_E	25,975.00	-0.00	None	24.00	30.00
5202	Level1	Level2	Pipeline_2	2		TGF_F	72,565.00	-0.00	None	78.00	30.00
20076	Level1	Level2	Pipeline_5	5		TGF_A	29,353.00	0.01	None	64.00	18.00

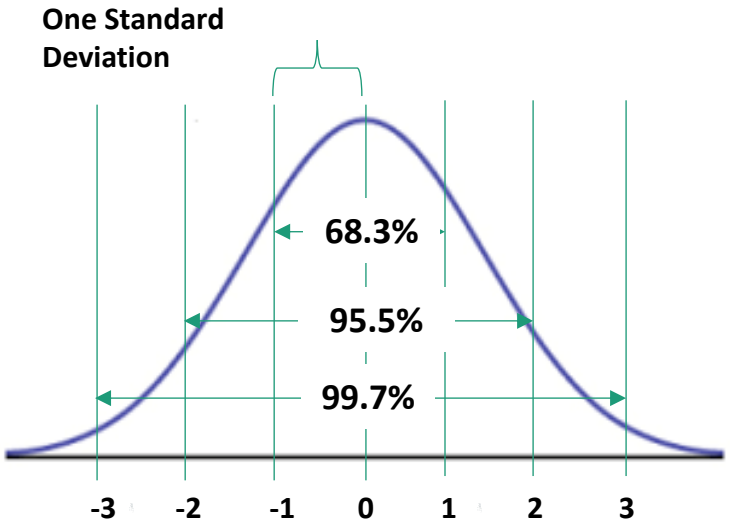
Predictors are Random

Applicability Learning Data

Target is Probability of Learning Data "Applicability"

Basic Statistics

Empirical Rule for Normal Distribution of Continuous Random Data



Population Mean μ

Sample Mean $\bar{x}$

Measures of Central Tendency
Mean, Median, Mode

Population Parameters

$$\text{standard deviation } \sigma = \sqrt{\frac{\sum(x-\mu)^2}{N}}$$

$$\text{variance} = \sigma^2$$

$$\text{population} = N$$

Sample Statistics

$$\text{standard deviation s} = \sqrt{\frac{\sum(x-\bar{x})^2}{n-1}}$$

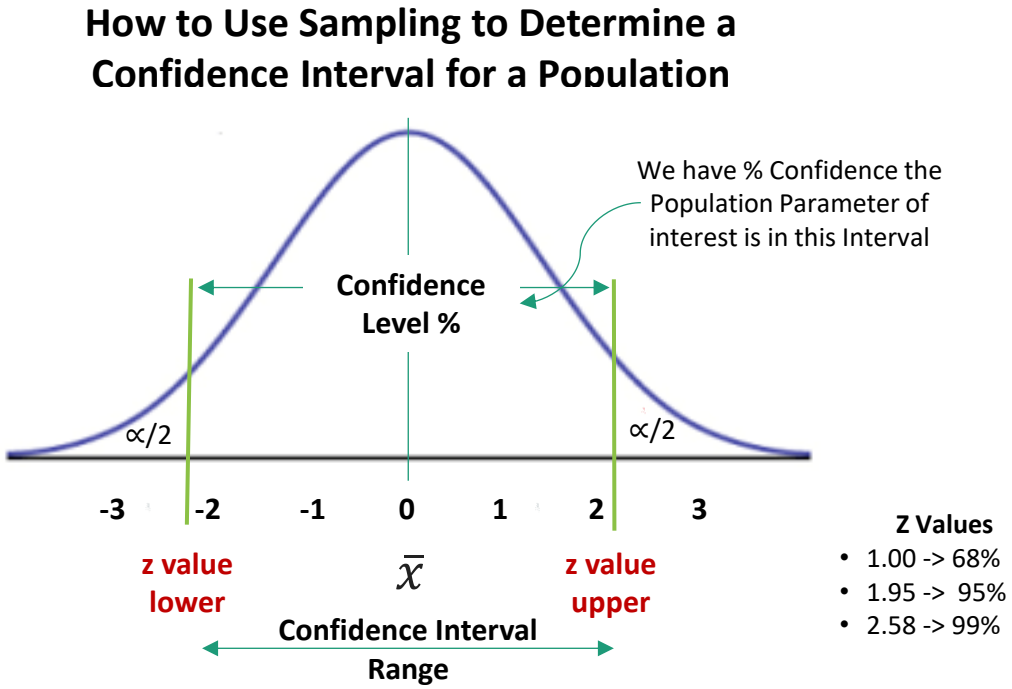
$$\text{variance} = s^2$$

$$\text{sample size} = n$$

Gaussian & Normal Distribution Function

$$P(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Confidence Intervals



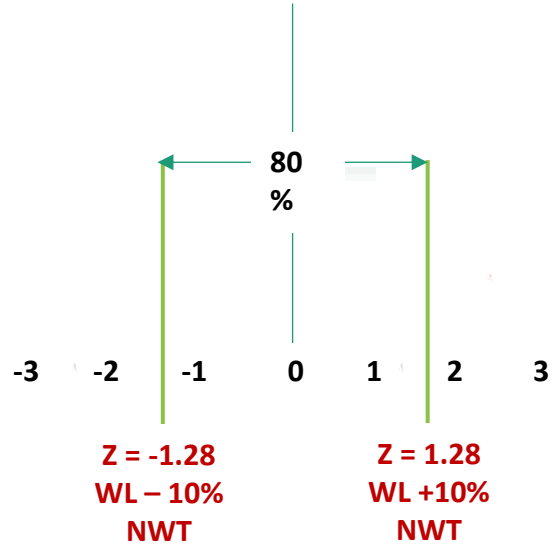
- Confidence Interval = $\bar{x} \pm$ critical $z \cdot$ Standard Error (SE)

where:

- Critical z from Standard Normal Distribution Table for required α (Level of Significance)
- SE $= \frac{\sigma}{\sqrt{n}}$ or approximately $\frac{s}{\sqrt{n}}$, that is, SE = average variance of sample means
- Margin of Error is $\pm$ z $\cdot$ SE

Confidence Intervals

How to Use Sampling to Determine a Confidence Interval for a Population Parameter



We have 80% Confidence the Population Mean (Belief) is in this Range

Working Example

The performance of in-line inspection tools are often characterized by criteria such as “accuracy of reported defects are +/- 10% of nominal wall thickness 80% of the time”.

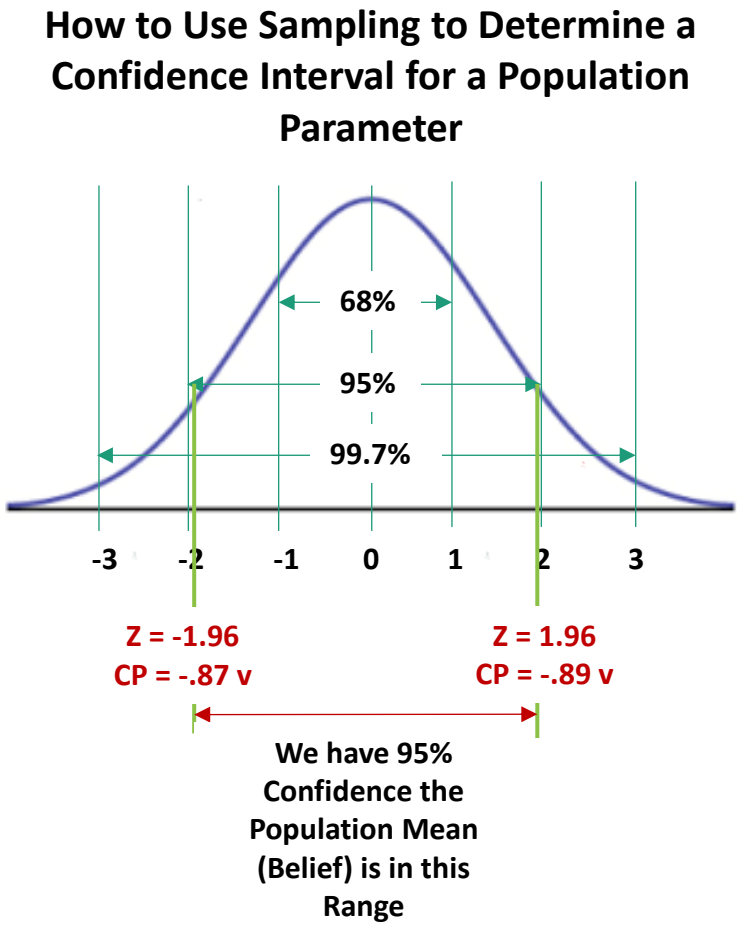
Assuming a normal distribution of reported defects, this means 80% of the defects would fall within ~1.28 Standard Deviations of the Mean, where 1.28 is the Z value on either side of the mean.

How is this Confidence Interval calculated?

Example:

- Reported defects should fall within +/- 10% NWT 80% of the Time
- Solve SE for 1 SD
- Z for 80% of Time = +/- 1.28 from Standard Normal Distribution Table
- +/- $10\% = 0 \pm 1.28$ x SE, where 0 is average error
- Solve for SE = .078
- The margin of error is $\pm$ z x SE = 1.28 x .078 = +/- 10%
- i.e. a 30% anomaly could range between 20-40% 80% of time

Confidence Intervals



Working Example

Using a representative population sample, we want to know what range of CP values include the cp population mean 95% of the time. We do not know the mean but would like to know the range where it is included 95% of the time. This will give us an idea of the state of cp across the broader population.

1. Collect Inputs

- Sample Size n = 30 (min. for Central Limit Theorem, Follows Normal Dist.)
- Sample Mean $\bar{x}$ = -.88 v (CP Off Reading Example)
- Sample Standard Deviation s = .03
- Sample Standard Error SE $= \frac{\sigma}{\sqrt{n}}$ or approximately $\frac{s}{\sqrt{n}}$ = .0055 v

2. Specify Required Confidence Level

- Specify Level of Significance $\alpha = .05$ (i.e., mean is not in 95% CI)
- Confidence Level = 1 - α = .95

3. Calculate Confidence Interval

- Find critical z-value in Stats Table for $\alpha/2$ = .025 (i.e., 2.5% for each left & right tail outside of 95% Confidence)
- z = $\pm$ 1.96)
- Confidence interval is $\bar{x} \pm z \cdot \mathrm{SE}$ = -.88 $\pm$ 1.96 · .0055 = -.87 to -.89
- The margin of error is $\pm$ 1.96 · .0055 or $\pm$ .01

Missing Data Mitigation

Missing or Zero Numerical Data

- Exclude
- Find Proxy
- Machine Learn Value
- Use Average or Representative Value

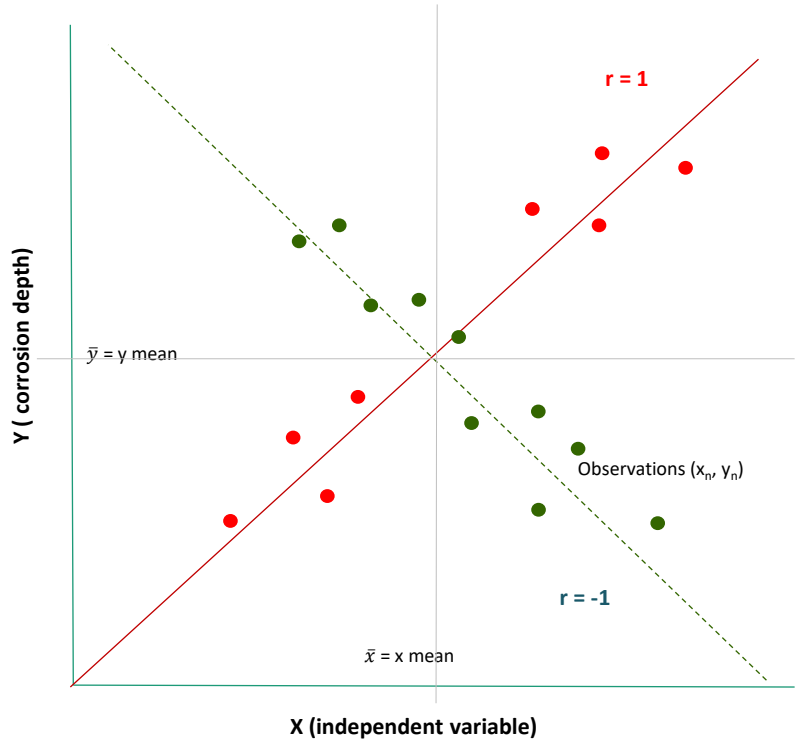
Categorical

- Exclude
- Find Proxy
- Machine Learn Value
- Default as “No_Data” Attribute
- Use Average or Representative Value

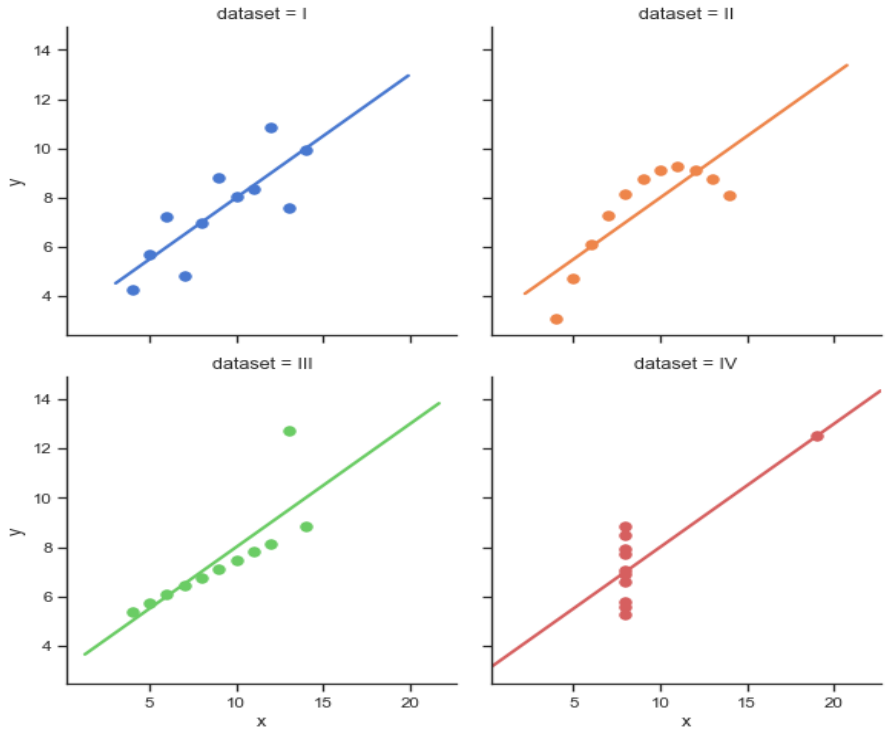
Correlation can be Misleading

r = Correlation Coefficient

Where r = -1 to 1



Anscombe's Quartet



$$r = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^2 \Sigma(y-\bar{y})^2}}$$

Correlation is any statistical association, though it commonly refers to the degree to which a pair of variables are linearly related