CSCE 590-1: Trusted AI

Lecture 5: AI: Supervised Machine Learning – Cont.

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

2ND SEP, 2021

Carolinian Creed: "I will practice personal and academic integrity."

Organization of Lecture 5

- Introduction Segment
  - Recap from Lecture 4
  - Project discussion

- Main Segment
  - Method: Regression Methods, Decision trees, Random Forest
  - Comparing methods
  - Open Data: Water

- Concluding Segment
  - About next lecture – Lecture 6
  - Ask me anything

Introductory Segment

Recap of Lecture 4

- Project sharing instructions and coding guidelines

- Did an overview of Machine learning
  - Looked at terminology
  - Type of methods: supervised, unsupervised and connections to type of analysis – descriptive, predictive and prescriptive
  - Metrics for evaluation

- Worked with COVID data

- Looked at basic method – linear regression

Project Discussion

- Information to be shared by students
  - Go to Google sheet: https://docs.google.com/spreadsheets/d/1VAX8ntb5zBQ-vOdsMHMhvEdwoaCZtuBaO4kJdkSA4eQ/edit?usp=sharing

  - Create a Google drive called "CSCE 590-1 Trusted AI (<YourName>)" and share with instructor: firstname.lastname@gmail.com
    - Put shared url in Column E
    - Put project title in column G
    - Create a folder in shared directory call project. Under it, have a Google doc called "Project Description". In it, have the following as bullets with associated details: Problem, User, AI Method, Data, Reliability: Testing, Holding Human Values, Human-AI interaction. See next slide for framework and guidance on what to put.

  - Put Github location for your code in F
    - Create one repository
    - For each quiz, project, etc, create a sub-folder

Course Project

- **Framework**
  1. (Problem) Think of a problem whose solution may benefit people (e.g., health, water, air, traffic, safety)
  2. (User) Consider how the primary user (e.g., patient, traveler) may be solving the problem today
  3. (AI Method) Think of what the solution will do to help the primary user
  4. (Data) Explore the data for a solution to work
  5. (Reliability:Testing) Think of the evaluation metric we should employ to establish that the solution will works? (e.g., 20% reduction in patient deaths)
  6. (Holding Human Values) Discuss if there are fairness/bias, privacy issues?
  7. (Human-AI) Finally, elaborate how you will explain the primary user that your solution is trustable to be used by them

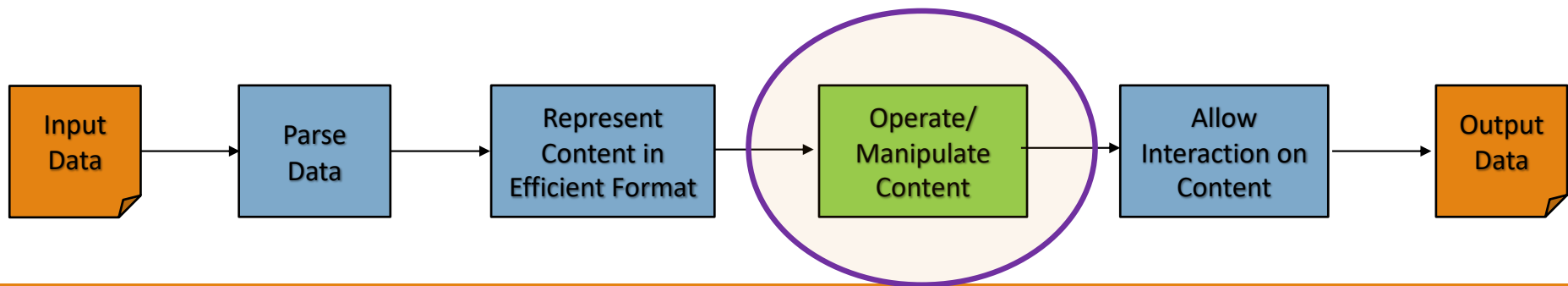# Rubric for Evaluation of Course Project

Project

- Project plan along framework introduced (7 points)
- Challenging nature of project
- Actual achievement
- Report
- Sharing of code

Presentation

- Motivation
- Coverage of related work
- Results and significance
- Handling of questions

Main Segment

Input Data → Parse Data → Represent Content in Efficient Format → Operate/Manipulate Content → Allow Interaction on Content → Output Data

Linear Methods

Assumption: target value (y) is expected to be a linear combination of the features (Xj).

Function estimate (linear)
W: weight, b: bias

$$f(X_j) = X_j W + b$$

Error Term (mean squared error)

$$MSE = \frac{1}{n} \sum_{j=1}^{n} \left[f(X_{j.}) - y_j \right]^2$$

Many variants depending on the nature of error being minimized: overfitting (Ridge), number of non-zero coefficients (Lasso), …
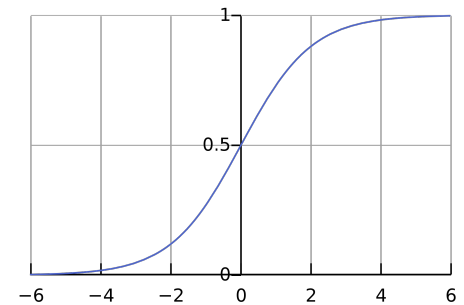
- Reference: https://scikit-learn.org/stable/modules/linear_model.html

Relationship Between Linear Regression and Classification

- Model type
  - Regression – Linear Regression:
    - predicting a continuous valued attribute assuming linear combination of feature vectors
  - Classification – Logistic Regression
    - Classifying a categorical attribute assuming linear combination of feature vectors
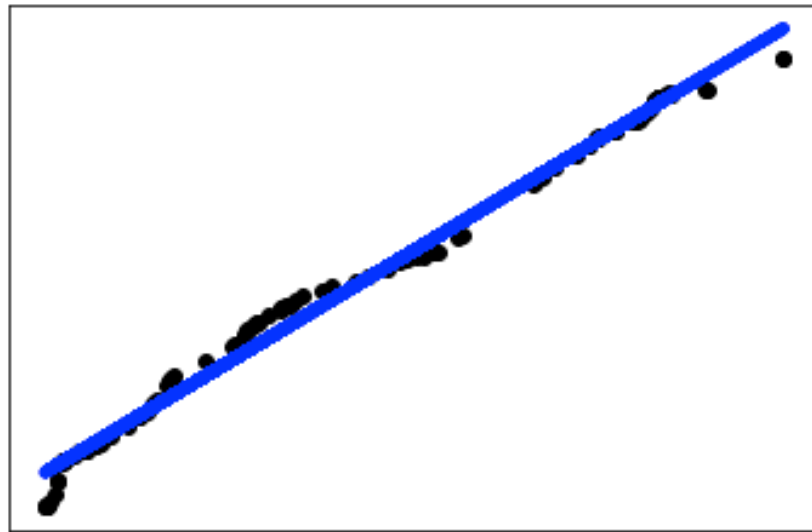
- Logit function

  Example*: t is a linear function of a single explanatory variable x*

  $$p(x) = \sigma(t) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

  Source: https://en.wikipedia.org/wiki/Logistic_regression
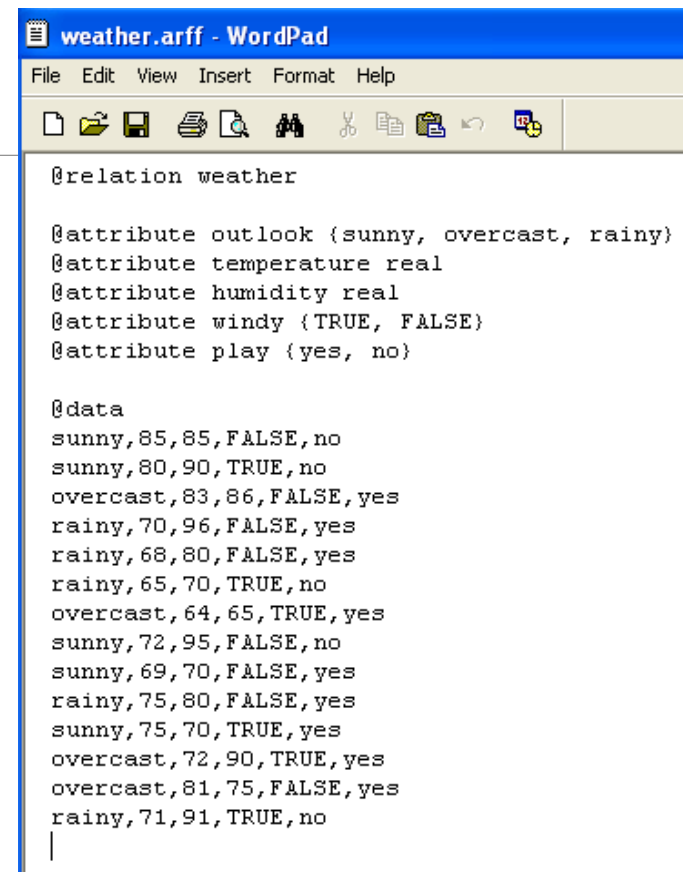
Linear Regression

Notebook: https://github.com/biplav-s/course-tai/blob/main/sample-code/l4-l5-supervised-ml/Supervised-Regression-Classification.ipynb

Reference and Demo

- Data: UCI Datasets - https://archive.ics.uci.edu/ml/datasets.php

- Tools:
  - Weka - https://www.cs.waikato.ac.nz/ml/weka/
    - ARFF format – Used by WEKA

ARFF Data Format

- Attribute-Relation File Format

- Header – describing the attribute types

- Data – (instances, examples) comma-separated list


```
@relation weather

@attribute outlook {sunny, overcast, rainy}
@attribute temperature real
@attribute humidity real
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
sunny,72,95,FALSE,no
sunny,69,70,FALSE,yes
rainy,75,80,FALSE,yes
sunny,75,70,TRUE,yes
overcast,72,90,TRUE,yes
overcast,81,75,FALSE,yes
rainy,71,91,TRUE,no
```

Decision Tree

Problem: Classify Weather Data

Input

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
...	...	...	...	...

Class Label

Output (Informal)

```
If outlook = sunny and humidity = high then play = no
If outlook = rainy and windy = true then play = no
If outlook = overcast then play = yes
If humidity = normal then play = yes
If none of the above then play = yes
```

Which Variable to Learn to Create Rules On?

- What do we want?
  - Compact model (e.g., set of rules)
  - High accuracy / low error

- Find the most discriminating variable
  - But how do we measure this

- Corner cases
  - If all the samples are the same, the decision tree is a ?
    - Leaf node with the only class
  - If there are no attributes in the dataset, the decision tree is?
    - A node with most common class

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
...	...	...	...	...

Expected Information/ Entropy

- Concept: Expected Information
  - Let
    - Class label has **m** distinct values (i.e., m distinct classes)
    - s_i be the number of samples of S of Class C_i (i = 1 ..**m)**
  - $I\ (s_1, s_2, .., s_m) = -\ \Sigma_{i=1\ to\ m}\ p_i\ \log_2\ (p_i)$
    - Where P_i is the probability a sample belongs to class Ci; estimated by(s_i/s)

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
...	...	...	...	...

- Entropy / Expected Information after partitioning on Attribute A which has **v** distinct values
  - $E(A) = \Sigma_{j=1\ to\ v}\ (s_{1j} + ... + s_{mj}\ )\ /\ S\ *\ (I\ (s_{1j}, s_{2j}, .., s_{mj})$
  - s_{ij} be the number of samples in S_j of Class C_i (i = 1 ..**m)**
  - Smaller the entropy, the greater the purity of the subset partitions
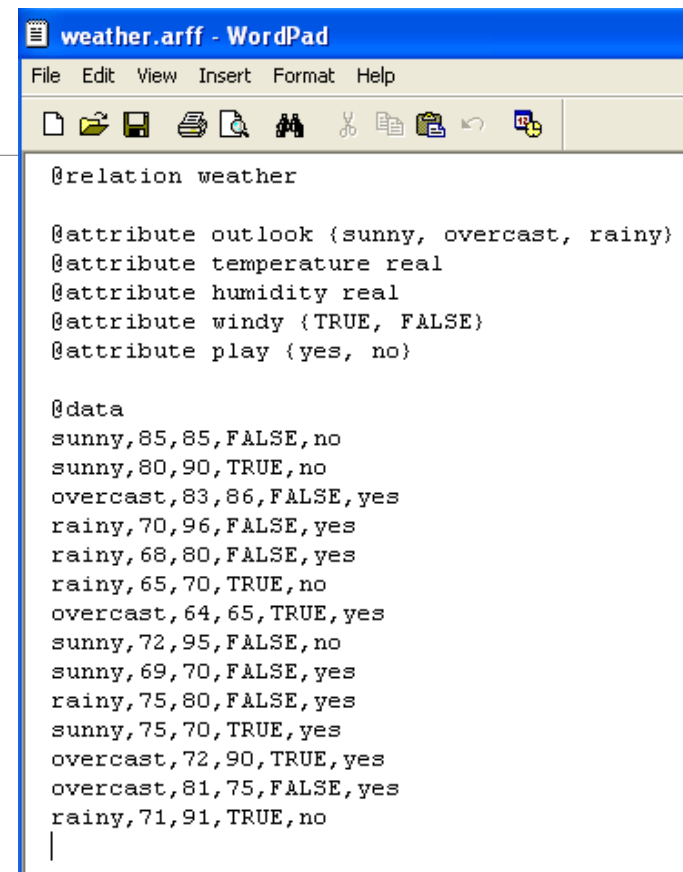
Information Gain

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
...	...	...	...	...

- Entropy / Expected Information after partitioning on Attribute A which has v distinct values
  - $E(A) = \Sigma_{j=1 \text{ to } v} (s_{1j} + ... + s_{mj}) / S \ * (I(s_{1j}, s_{2j}, .., s_{mj})$
    - s_{ij} be the number of samples in S_j of Class C_i (i = 1 ..m)

- After partition, S_j
  - $I(s_{1j}, s_{2j}, .., s_{mj}) = - \Sigma_{i=1 \text{ to } m} p_{ij} \log_2 (p_{ij})$
  - Where p_{ij} is the probability a sample in S_j belongs to class C_i; estimated by $(s_{ij} / |s_j|)$

- Gain (A) = $I(s_1, s_2, .., s_m) - E(A)$
  - Is the expected reduction in entropy by knowing the value of Attribute A

- **Method**: Split on the attribute which leads to the highest information gain

Weka Exercise

ARFF Data Format

- Data is in ARFF in UCI dataset

- Or Convert
  - File system, CSV → ARFF format
  - Use C45Loader and CSVLoader to convert

weather.arff - WordPad

File Edit View Insert Format Help

```
@relation weather

@attribute outlook {sunny, overcast, rainy}
@attribute temperature real
@attribute humidity real
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
sunny,72,95,FALSE,no
sunny,69,70,FALSE,yes
rainy,75,80,FALSE,yes
sunny,75,70,TRUE,yes
overcast,72,90,TRUE,yes
overcast,81,75,FALSE,yes
rainy,71,91,TRUE,no
```
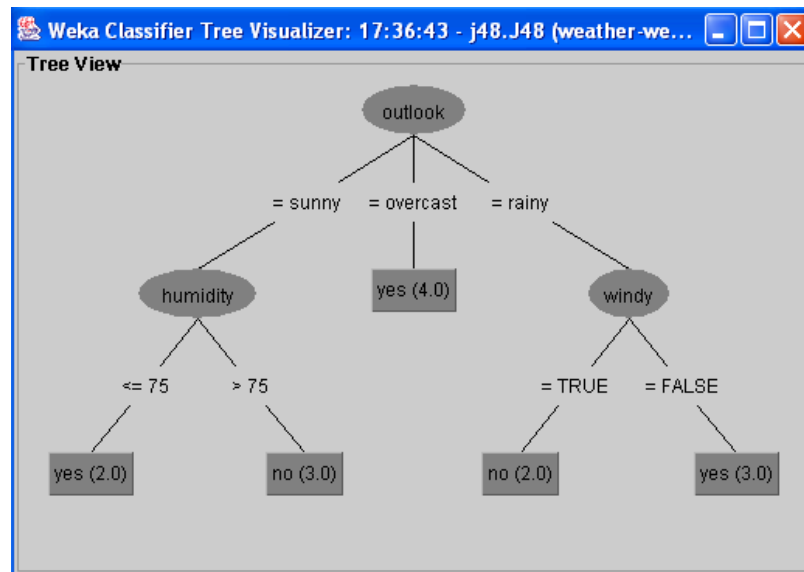
Weka: weka.classifiers.trees.J48

Class for generating an unpruned or a
pruned C4.5 decision tree.
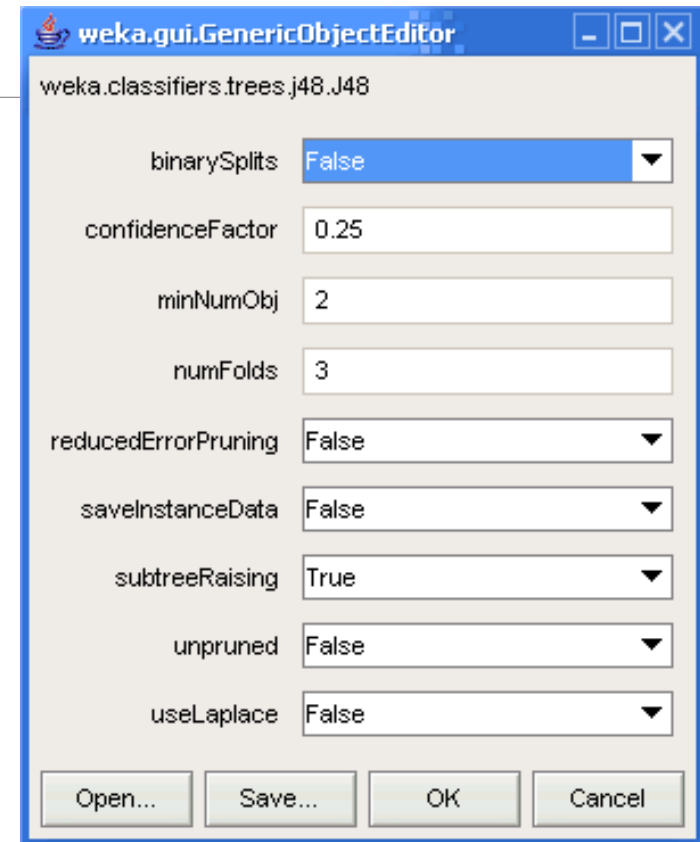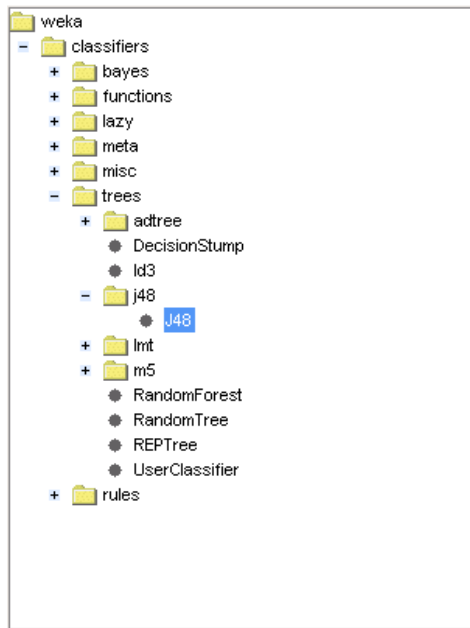
Understanding Output

Weka: Decision Tree Output

J48 pruned tree
----

outlook = sunny
| humidity = high: no (3.0)
| humidity = normal: yes (2.0)
outlook = overcast: yes (4.0)
outlook = rainy
| windy = TRUE: no (2.0)
| windy = FALSE: yes (3.0)

Number of Leaves :   5

Size of the tree :   8

=== Summary ===

Correctly Classified Instances   7   50   %
Incorrectly Classified Instances   7   50   %
Kappa statistic   -0.0426
Mean absolute error   0.4167
Root mean squared error   0.5984
Relative absolute error   87.5   %
Root relative squared error   121.2987 %
Total Number of Instances   14

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.556	0.6	0.625	0.556	0.588	0.633	yes
	0.4	0.444	0.333	0.4	0.364	0.633	no
Weighted Avg.	0.5	0.544	0.521	0.5	0.508	0.633	

=== Confusion Matrix ===

 a b <-- classified as
 5 4 | a = yes
 3 2 | b = no

Test Options

- Percentage Split (2/3 Training; 1/3 Testing)

- Cross-validation
  - Estimating the generalization error based on resampling when limited data
    - averaged error estimate.
  - Cross-fold validation (10-fold)
  - Leave-one-out (Loo)
  - Stratified

Comparing Classification Methods

- Predictive accuracy

- Interpretability: providing insight

- Robustness: handling noisy data

- Speed

- Scalability: large volume of data

References

- Blogs: https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/

- Google: https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

- Insead:
  - Description: https://inseaddataanalytics.github.io/INSEADAnalytics/CourseSessions/Sessions67/ClassificationAnalysisReading.html
  - Data analytics for Business: https://inseaddataanalytics.github.io/INSEADAnalytics/

Water Data – Water Atlas

- Data download:
https://dev.chnep.wateratlas.usf.edu/data-download/beta/

- Local cache:
https://github.com/biplav-s/course-tai/tree/main/sample-code/common-data/water

Water – Use Cases

- "Water quality status and trends is always a big issue. There are dozens of separate agencies who collect water quality data, different labs – NELAC certified and not, and different formats and repositories of data. In order to determine the status or trend of wq for a waterbody, you have to collect a sufficiently large dataset, but also minimize the presence of bad/questionable data. This is an ongoing challenge for us. In Florida, all local governments and agencies are required to upload data to the WIN (a STORET replacement) database if they want their data used for Impaired/TMDL determinations. We have even found bad data in WIN that were actually collected by FDEP. Figuring out how to minimize "bad" data in the absence of robust data qualifiers might be a nice use-case.

- Along those lines, we implemented a WQ Trends analysis using seasonal Kendall Tau. For Sarasota County (https://www.sarasota.wateratlas.usf.edu/water-quality-trends/), we only use the County monitoring data because they have very rigorous procedures (and Mote does their monitoring and lab analysis). For CHNEP (https://chnep.wateratlas.usf.edu/water-quality-trends/), they include data from numerous sources. We built in filters to remove data based on qa codes, but I always question whether there might be bad data without qa codes. Maybe your students want to develop an AI approach to the "trend analysis" and compare the results to the stats we generate. The data and the R script used for the analysis is available on the web pages.

- People obviously want to know if it safe to go to the beach or swim in the water. Once again, we have several agencies who conduct monitoring for bacteria and red tide. We built the Recreational Water Quality Map to pull together the various monitoring results, but the map itself does not give people an answer – it only provides the data that they then have to interpret. Perhaps the students can use AI to provide a clearer answer. https://www.pinellas.wateratlas.usf.edu/maps/coastal-water-quality-map/"

References

- Blogs: https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/

- Google: https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

- Insead:
  - Description: https://inseaddataanalytics.github.io/INSEADAnalytics/CourseSessions/Sessions67/ClassificationAnalysisReading.html
  - Data analytics for Business: https://inseaddataanalytics.github.io/INSEADAnalytics/

Machine Learning – Insights from Data

- Descriptive analysis
  - Describe a past phenomenon
  - **Methods**: classification, clustering, dimensionality reduction, anomaly detection, neural methods

- Predictive analysis
  - Predict about a new situation
  - **Methods**: time-series, neural networks

- Prescriptive analysis
  - What an agent should do
  - **Methods**: simulation, reinforcement learning, reasoning

- New areas
  - Counterfactual analysis
  - Causal Inferencing
  - Scenario planning

Concluding Segment

Lecture 5: Concluding Comments

- We completed discussion of linear methods

- Explored decision trees

- Looked at water data

About Next Lecture – Lecture 6

Lecture 6: Supervised ML – Trust Issues

- We will look at random forest

- Trust issues with supervised ML
  - Reading material
    - Turing Institute report on AI for Covid in UK (File: Report-data-science-and-ai-in-the-age-of-covid_full-report_2.pdf)
    - Bias in AI. (File: Article-MLBias-Practice-CACM-Aug2021.pdf) – *following class*

- Quiz1