Weekly meeting 7

Dr. Doina Bein
Thursday, June 29, 10:00am-11am

Surveys to be completed

To be done today, before starting research:

CIC-PCUBED Pre-event survey:

https://fullerton.qualtrics.com/jfe/form/SV_6YIVSkC6hLxbunA

Project 1: Data Science

What you need to do: topics & objectives

Objective 1: Learn Python using some textbook or some online courses such as (https://www.codecademy.com/learn/learn-python). Shared by Stephanie Pocci: Learn Python in a couple hours. This YouTuber does a very beginner-friendly crash course about the capabilities of Python and its uses. Here is the link: https://www.youtube.com/watch?v=rfscVS0vtbw

Objective 2: Learn how to use Jupyter Notebook. Start here http://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html

Objective 3: For data science, find a suitable dataset and start training some neural network using with Google tensorflow.

Logistics for all students

- Who is participating: list of current research students and their availability
- Research will be conducted virtually during the week with in-person meetings throughout the week
- Zoom meetings for me to teach new topics and for you to participate in open discussions
- Support:
  - If needed, you can meet me
    Zoom: Mon, Tu, Wed from 8:30-10:25 am
    IN PERSON: Mon, Tu, Wed from 8:30-9:30 am, Thursday 8:30-10am or by email
  - CIC-PCUBED peer mentor: (tentative) availability

Logistics for all students (contd.)

- Make a copy of this GDoc Work schedule, share the Gdoc copy with me, and maintain it weekly and daily; due at the end of Week 2
- Before the end of week 3, make a copy of this GDoc Proposed work and maintain your copy by individual or teams of up to three; due by the end of Week 3
- Complete your availability here; try to have it consistent over the 7 weeks such that it will be easy to partner in the project
- Group projects: to be decided; sample list here
- Oral or poster presentations: tentatively scheduled for Friday, July 28, from 8:30am-12:30 pm and if needed, from 1:30-4 pm

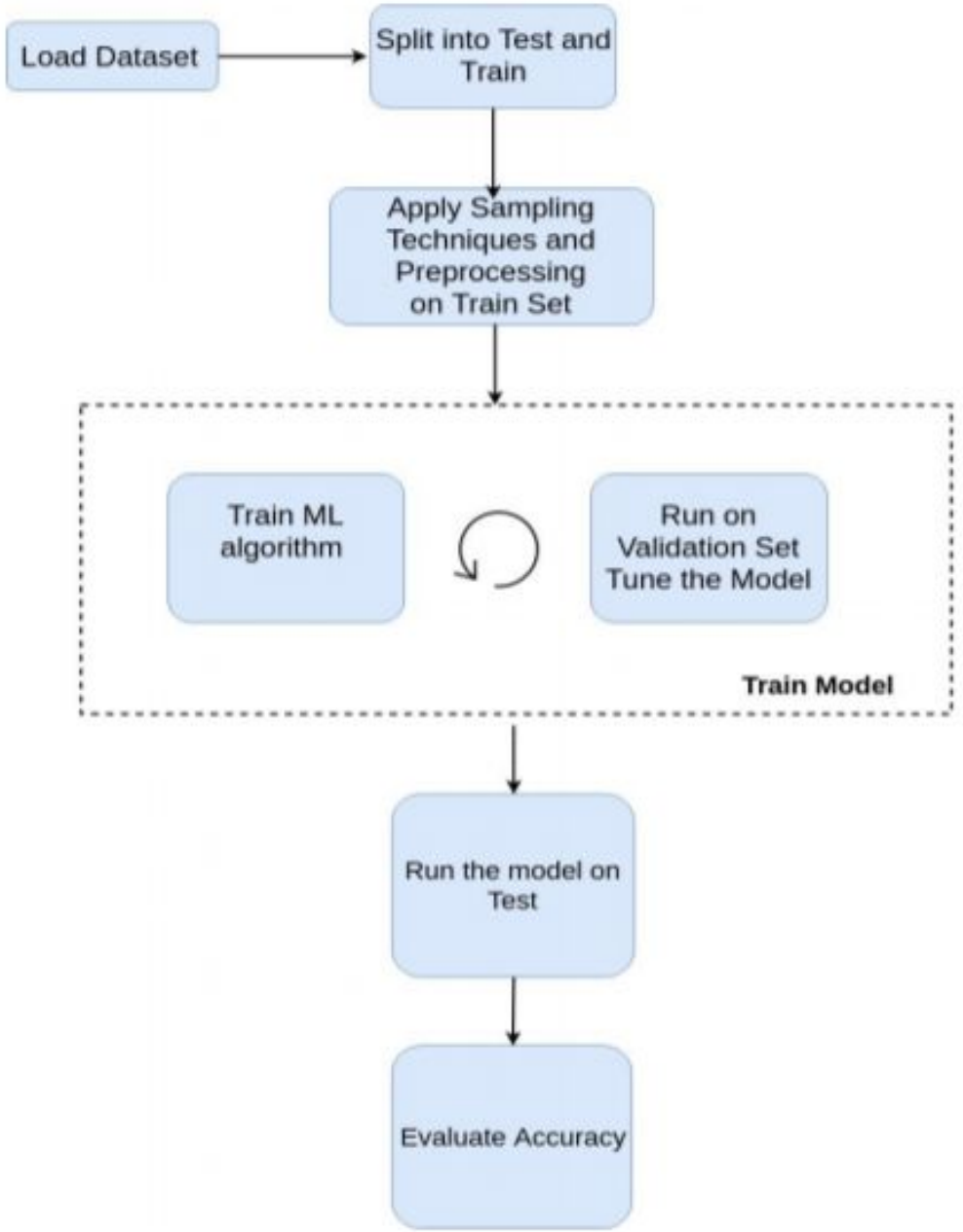
Please checkout:

- Other websites and ebooks
- Websites with free datasets
- More resources on selected topics
- If you find good, free resources, please share it by email or during weekly meetings
- Next meeting: I will lecture on ZOOM on Data Science: Monday, July 3, from 10:30am-12pm

Progress on Learning Python

- Free course: https://www.codecademy.com/learn/learn-python
- Free course: https://www.kaggle.com/learn/python
- Youtube video (about 4 hours): https://www.youtube.com/watch?v=rfscVS0vtbw

Data Science


Load Dataset
Split into Test and Train
Apply Sampling Techniques and Preprocessing on Train Set
Train ML algorithm
Run on Validation Set Tune the Model
Train Model
Run the model on Test
Evaluate Accuracy

Obtaining your free dataset

- Check this GDoc created by me in Summer 2021: https://docs.google.com/document/d/1bSFxrX0_PdugEuv6s7GWpS1M_7yxmUuVy87VhSyR1po/edit?usp=sharing
- Demo on how to search on Kaggle and Data.gov

Evaluating the ML Model

- The test data will be used to evaluate if model has learnt correctly.
- Model’s performance is measured and its accuracy evaluated.
- The performance measures adapted in this model are Area under ROC, Precision, Recall, and Average Precision.

Classifiers

(storm.cis.fordham.edu/~gweiss/ubdm05/Holte-slides.ppt)

- A *classifier* assigns an object to one of a predefined set of categories or classes.
- Examples:
  - A metal detector either sounds an alarm or stays quiet when someone walks through.
  - A credit card application is either approved or denied.
  - A medical test's outcome is either positive or negative.
- This talk: only two classes, "positive" and "negative".

Two Types of Error

(storm.cis.fordham.edu/~gweiss/ubdm05/Holte-slides.ppt)

False positive ("false alarm"), FP

alarm sounds but person is not carrying metal

False negative ("miss"), FN

alarm doesn't sound but person is carrying metal

Confusion Matrix

(www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html)

- A confusion matrix (Kohavi and Provost, 1998) contains information about actual and predicted classifications done by a classification system. Performance of such systems is commonly evaluated using the data in the matrix. The following table shows the confusion matrix for a two class classifier.

a	b
c	d

- The entries in the confusion matrix have the following meaning in the context of our study:
- a is the number of correct predictions that an instance is negative,
- b is the number of incorrect predictions that an instance is positive,
- c is the number of incorrect of predictions that an instance negative, and
- d is the number of correct predictions that an instance is positive.

Confusion Matrices and True/False Positive/Negative

(www.washburn.edu/faculty/boncella/.../Lecture%204%20-%20Model%20Evaluation....)

- The confusion matrix $M = [m_{i,j}]$ stores its coefficients $m_{i,j}$ of well-classified rates when x is classified as C_i (estimated class) with the ground-truth class being C_j :

Classification Confusion Matrix		
	Predicted Class	
Actual Class	1	0
1	201	85
0	25	2689

201 1's correctly classified as "1"

85 1's incorrectly classified as "0"

25 0's incorrectly classified as "1"

2689 0's correctly classified as "0"

Error Rate

(www.washburn.edu/faculty/boncella/.../Lecture%204%20-%20Model%20Evaluation....)

Classification Confusion Matrix		
	Predicted Class	
Actual Class	1	0
1	201	85
0	25	2689

Overall error rate = (25+85)/3000 = 3.67%

Accuracy = 1 – err = (201+2689) = 96.33%

If multiple classes, error rate is:

(sum of misclassified data)/(total data)

Example: 3 classifiers

(storm.cis.fordham.edu/~gweiss/ubdm05/Holte-slides.ppt)

True	Predicted	
	pos	neg
pos	40	60
neg	30	70

True	Predicted	
	pos	neg
pos	70	30
neg	50	50

True	Predicted	
	pos	neg
pos	60	40
neg	20	80

Classifier 1
TP = 40
FP = 30

Classifier 2
TP = 70
FP = 50

Classifier 3
TP = 60
FP = 20

Assumptions

(storm.cis.fordham.edu/~gweiss/ubdm05/Holte-slides.ppt)

- Standard Cost Model
  - correct classification costs 0
  - cost of misclassification depends only on the class, not on the individual example
  - over a set of examples costs are additive
- Costs or Class Distributions:
  - are not known precisely at evaluation time
  - may vary with time
  - may depend on where the classifier is deployed
- True FP and TP do not vary with time or location, and are accurately estimated.

How to Evaluate Performance ?

(storm.cis.fordham.edu/~gweiss/ubdm05/Holte-slides.ppt)

- Scalar Measures
  - Accuracy
  - Expected cost
  - Area under the ROC curve
- Visualization Techniques
  - ROC curves
  - Cost Curves

Accuracy, Error Rate, Sensitivity and Specificity

(hanj.cs.illinois.edu/bk3/bk3_slides/08ClassBasic.ppt)

A\P	C	¬C	
C	**TP**	**FN**	**P**
¬C	**FP**	**TN**	**N**
	P'	**N'**	**All**

- **Classifier Accuracy,** or recognition rate: percentage of test set tuples that are correctly classified

  Accuracy = (TP + TN)/All

- **Error rate:** *1 – accuracy*, or

  Error rate = (FP + FN)/All

- **Class Imbalance Problem**:
  - One class may be *rare*, e.g. fraud, or HIV-positive
  - Significant *majority of the negative class* and minority of the positive class
  - **Sensitivity**: True Positive recognition rate
    - **Sensitivity = TP/P**
  - **Specificity**: True Negative recognition rate
    - **Specificity = TN/N**

Precision and Recall, and F-measures

(hanj.cs.illinois.edu/bk3/bk3_slides/08ClassBasic.ppt)

- **Precision**: exactness – what % of tuples that the classifier labeled as positive are actually positive

$$precision = \frac{TP}{TP + FP}$$

- **Recall:** completeness – what % of positive tuples did the classifier label as positive?
- Perfect score is 1.0
- Inverse relationship between precision & recall
- ***F* measure (*F*$_1$ or *F*-score)**: harmonic mean of precision and recall,

$$recall = \frac{TP}{TP + FN}$$

$$F = \frac{2 \times precision \times recall}{precision + recall}$$

- F_β: weighted measure of precision and recall
  - assigns ß times as much weight to recall as to precision

$$F_\beta = \frac{(1 + \beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$$

Classifier Evaluation Metrics: Example

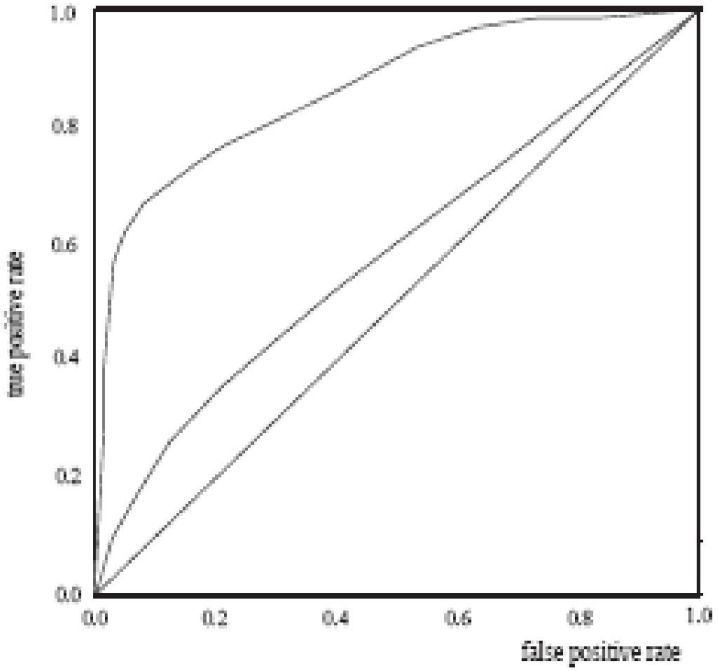
(hanj.cs.illinois.edu/bk3/bk3_slides/08ClassBasic.ppt)

Actual Class\Predicted class	cancer = yes	cancer = no	Total	Recognition(%)
cancer = yes	**90**	**210**	300	30.00 (*sensitivity)*
cancer = no	**140**	**9560**	9700	98.56 (*specificity)*
Total	230	9770	10000	96.50 (*accuracy*)

- *Precision* = 90/230 = 39.13% *Recall* = 90/300 = 30.00%

Model Selection: ROC Curves

- **ROC** (Receiver Operating Characteristics) curves: for visual comparison of classification models
- Originated from signal detection theory
- Shows the trade-off between the true positive rate and the false positive rate
- The area under the ROC curve is a measure of the accuracy of the model
- Rank the test tuples in decreasing order: the one that is most likely to belong to the positive class appears at the top of the list
- The closer to the diagonal line (i.e., the closer the area is to 0.5), the less accurate is the model



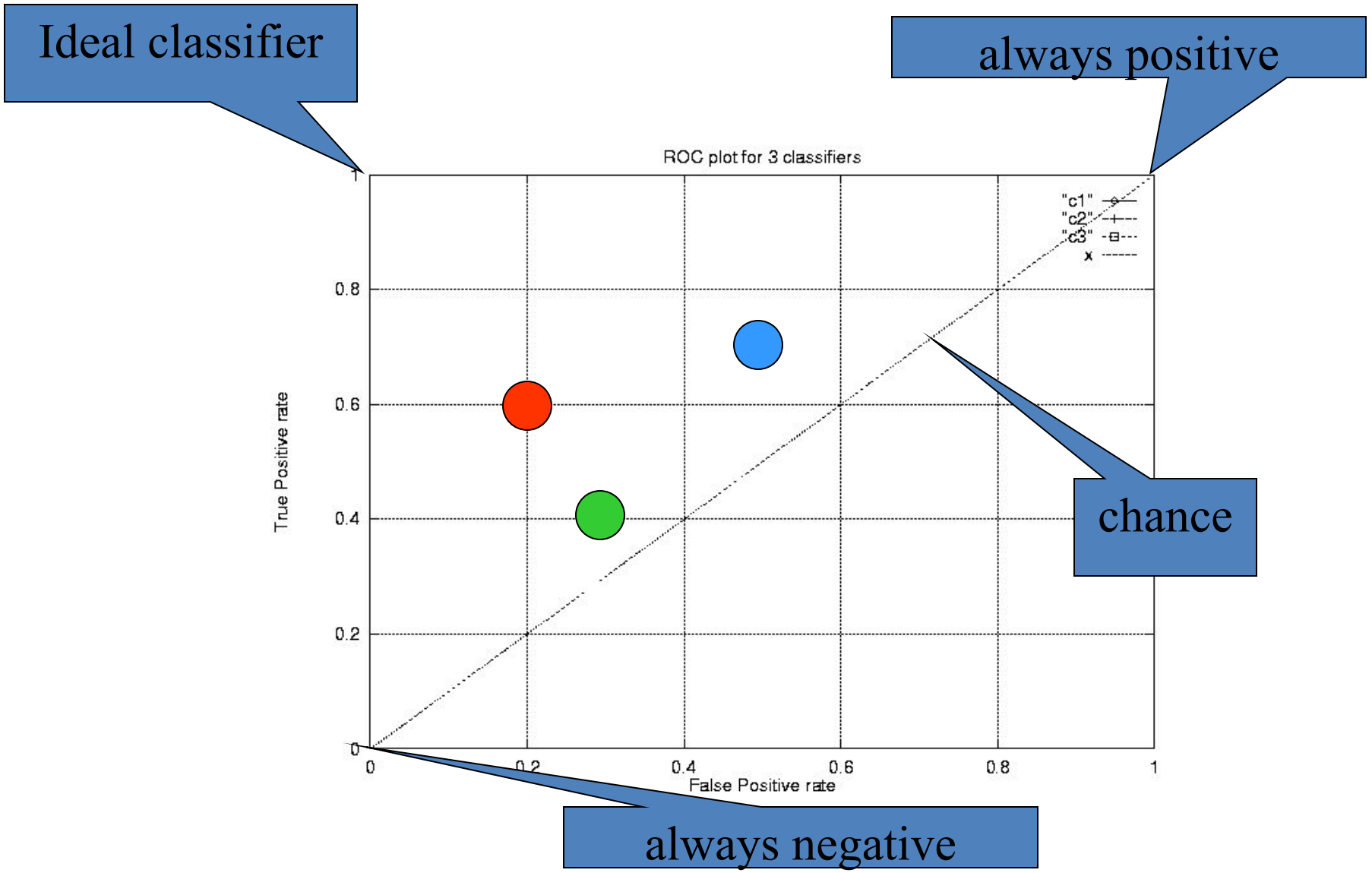
- Vertical axis represents the true positive rate
- Horizontal axis rep. the false positive rate
- The plot also shows a diagonal line
- A model with perfect accuracy will have an area of 1.0

ROC curves

- Area under curve gives idea of how good classifier is. 0.5 = no good, approaching 1 = excellent
- Can then build in profits/costs of different correct answers/mistakes into the confusion matrices to build a Gains Chart. Again, look at this area on chart
- Classifier with highest area on gains chart is the most profitable

ROC plot for the 3 Classifiers

(storm.cis.fordham.edu/~gweiss/ubdm05/Holte-slides.ppt)



When One Class is More Important

(www.washburn.edu/faculty/boncella/.../Lecture%204%20-%20Model%20Evaluation....)

Actual class\Predicted class	C_1	$\neg C_1$
C_1	**True Positives (TP)**	**False Negatives (FN)**
$\neg C_1$	**False Positives (FP)**	**True Negatives (TN)**

- In many cases it is more important to identify members of one class
  - Tax fraud
  - Credit default
  - Response to promotional offer
  - Detecting electronic network intrusion
  - Predicting delayed flights
- In such cases, we are willing to tolerate greater overall error, in return for better identifying the important class for further attention

Classification of Class-Imbalanced Data Sets

(hanj.cs.illinois.edu/bk3/bk3_slides/08ClassBasic.ppt)

- Class-imbalance problem: Rare positive example but numerous negative ones, e.g., medical diagnosis, fraud, oil-spill, fault, etc.
- Traditional methods assume a balanced distribution of classes and equal error costs: not suitable for class-imbalanced data
- Typical methods for imbalance data in 2-class classification:
  - **Oversampling**: re-sampling of data from positive class
  - **Under-sampling**: randomly eliminate tuples from negative class
  - **Threshold-moving**: moves the decision threshold, t, so that the rare class tuples are easier to classify, and hence, less chance of costly false negative errors
  - Ensemble techniques: Ensemble multiple classifiers introduced above
- Still difficult for class imbalance problem on multiclass tasks

Classification vs. Prediction

- Classification:
  - predicts categorical class labels
  - classifies data (constructs a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data
- Regression:
  - models continuous-valued functions, i.e., predicts unknown or missing values
- Typical Applications
  - credit approval
  - target marketing
  - medical diagnosis
  - treatment effectiveness analysis

What is expected of you

- Learn Python well enough to do a data science project
- Learn Jupyter Notebook well enough to be able to open a CSV file, read the data, analyze the relevant columns, call some function in the Python library to do some clustering and/or classification
- Display the results as numerical data
- Optional: plot the results using plotly

Round table discussion on topics

- We will discuss progress, findings, roadblocks, ideas how to solve problems without copying other people's projects from GitHub.
- Unlike industry where they give you a problem to solve, you choose the problem you want to solve.
- You need to apply one or more of the techniques I mentioned in my lectures.
- If you get good results, excellent.
- If you do not get good results, excellent too, at least you learn how to apply these methods.
- Do you have things to share: progress, roadblocks, interesting datasets, coding issues?