Machine Learning for Science & Society

- Spring 2021
- Tuesdays 5:30-8:15
- Professor Sarah Brown
- CSC 592: Topics in Computer Science

In this class, we will address the challenges in applying machine learning to scientific research and in high stakes social contexts. On the science side, we will examine the role of ML in research, in particular how it works within knowledge production and how to evaluate ML in line with domain norms. On the social side, we will consider how to ensure ML-based algorithmic decision making systems uphold social values, with a focus on fairness. While these two applications are distinct, many of the challenges translate into common technical problems. Some of the common challenges include:

- missing data
- noisy or missing labels
- multiple objectives

We will look at a range of strategies for identifying and mitigating these problems including:

- robust evaluation
- model inspection
- explanations
- interpretable models

Format

This will be a synchronous course offered via Zoom. To successfully participate in this course students will need:

- consistent internet access during class time
- to use URI SSO credentials to access readings and materials
- a microphone to participate in conversations in class, at least most sessions

The course will involve:

- reading and evaluating ML research papers
- facilitating and participating in class discussions of the papers
- producing a replication, demo, or illustration of one concept covered for a broader audience
- completing a project using ML in a scientific or social domain
- writing a CS conference style (short & concise) final paper on their project

graduate students are encouraged to do a project related to their research

Prerequisites

To be successful in this class students should have:

- past experience with machine (CSC461 or equivalent)
- basic programming skills
- familiarity with concepts in probability, linear algebra, and calculus that appear in ML

varying skill in these topics is ok, but a general understanding of the basic ideas is important. If you're interested and not sure if you have the background, complete the form below

Basic Facts

Meetings

This class will meet on Monday and Wednesday 3-4:15pm. via Zoom (link provided to registered students via BrightSpace)

Contents

> **Tip**
>
> You can [subscribe to the Brightspace Calendar] to access it in your favorite calendaring tool.

Instructor

Professor Sarah M Brown is an Assistant Professor in Computer Science. Her current research aims to answer the question, "How can machine learning produce AI systems that make fair decisions?"

Office Hours

By appointment, link on Brightspace.

Schedule

We will meet synchronously via Zoom: Tu 5:30-8:15

This course will proceed in three main parts: overview, deep dives, and wrap up.

Structure

Overview

In the first part of the course we will review ML basics, set norms for interaction and complete a survey of the topics that we will cover for the rest of the semester.

In this part of the class, Professor Brown will lead synchronous sessions. Students will be responsible for reading overviews, refreshing background material, and choosing an area for their course project. Students will start with an introductory demo or replication as a mini project.

Deep Dives

During the middle of the course we will spend one week on each topic. There will be 1-3 papers to read each week.

Students will be responsible for presenting papers in class on a rotating basis.

During this time students will have milestones where they need to complete interim steps for their course project. The first milestone will be a proposal that includes the specific products for the remainder of the milestones based on a template.

Conclusion

In the end of the course, we will focus on integrating ideas across multiple topics.

We will also workshop students' projects, giving substantive feedback prior to the final submissions.

Final projects will be evaluated through a presentation and paper

Weekly topics

Class	Topic	Reading	Activities
2021-01-29	Introduction	None	introductions, expectation setting
2021-02-01	Probability Review	Model Based ML, chapter 1	reading discussion, setting
2021-02-03	ML Process & Mutual information preview	Scikit learn getting started,	live coding
2021-02-08	Missing Data: Intro strategies	Handling Missing Values when Applying Classification Models & Missing data imputation using statistical and machine learning methods in a real breast cancer problem	Paper discussion led by Daniel
2021-02-10	Missing data with graphical models and causal reasoning	Graphical Models for Inference with Missing Data & Missing Data as a Causal and Probabilistic Problem	Paper discussion led by Julian
2021-02-15	Current Challenges in Missing data	Handling Missing Data in Decision Trees: A Probabilistic Approach & How to miss data? Reinforcement learning for environments with high observation cost	Paper discussions by Xavier and Zhen
2021-02-17	Current Challenges in Missing data	How to deal with missing data in supervised deep learning	Paper discussion by Madhukara, Replication & testing discussion,
2021-02-22	Fairness	fairml classification chapter and friedler empricial comparison paper	Empirical setup
2021-02-24	Fairness	Reading	preview of lasso and admm constraint to multiobjecitve reformulation
2021-03-01	Multi-objective & constrained opt	Elastic Net	Paper presentation by Daniel, try out elastic net & LASSO in scikit learn
2021-03-03	Multi-objective & constrained opt	A critical review of multi-objective optimization in data mining: a position paper	Paper presentation and discussion by Zhen
2021-03-08	Latent Variable Models	Gaussian Mixture Models and Topic Models	Paper presentaiton by Xavier
2021-03-10	Latent Variable Models	Indian Buffet Process and Auto-Encoding Variational Bayes	Paper presentation by Madhukara
2021-03-15	Missing or Noisy labels	Learning with Noisy Labels and Semi Supervised Learning	Julian and Daniel
2021-03-17	Noisy Labels as a model for Bias	Recovering from biased data: Can fairness constraints improve accuracy and Fair classification with group dependent label noise	Zhen
2021-03-22	Interpretable & Explanation Intro	A Survey of Methods for Explaining Black Box Models	Activities
2021-03-24	A Case for Interpretability over Explanation	Why are we explaining black box models and Learning Certifiably optimal rule lists for categorical data	Activities

Class	Topic	Reading	Activities
2021-03-29	Models for Explanation	TCAV	Activities
2021-03-31	Perturbation based	Falsifiable, survey	Activities
2021-04-05	What are the risks of explanations	Reading	Activities
2021-04-07	What does Interpretable mean	[Towards A Rigorous Science of Interpretable Machine Learning]()	Activities
2021-04-12	Meta issues	ml & phil sci	Activities
2021-04-13	Meta issues	Roles for computing in social change	Activities
2021-04-19	Project Presentations	projects	peer feedback
2021-04-21	Project Presentations	projects	peer feedback
2021-04-26	Project Presentations	Reading	revision plans

Table 1 Schedule

Learning Outcomes and Evaluation

This course has goals with respect to the knowledge and research skills.

Evaluation will be with respect to each of the outcomes and based on a level of mastery: general awareness, competency, or mastery.

By the end of the course students will be able to:

- Critique common ways that social or scientific applications of ML require violating ML algorithm assumptions and ways to mitigate or adapt.
- Evaluate ML Research papers for their applicability to scientific and social applications of ML.
- Communicate about ML and its limitations work to varied audiences
- Apply ML to scientific and social data responsibly

Activities

- reading and evaluating ML research papers
- facilitating and participating in class discussions of the papers
- producing a replication, demo, or illustration of one concept covered for a broader audience
- completing a project using ML in a scientific or social domain
- writing a CS conference style (short & concise) final paper on the project

Evaluation

The grading scheme is rooted in achieving the learning outcomes.

Presentations, Discussion, and Exercises (40%)

Presenting papers and participating in class will contribute to demonstrating a basic awareness at each of learning objective.

Each class session will be evaluated on specification if you contribute to discussion or not.

Each time you present (20%) will be evaluated on

Translation mini project (10%)

For this assignment you can choose any topic other than the one your project is for and produce a short demo, illustration, or replication that makes some aspect of the the topic accessible for a broader audience.

For this, you must submit a one paragraph proposal that describes your demo Once that's approved that it will count, you have two weeks to build your demo or replication. The latest your demo may be submitted is at the same time as your final project.

Your demo proposal must:

- state the topic from class your demo relates to
- state the format/medium your demo will take:
  - illustration, replication, interactive visualization, etc
- describe the target audience (a particular type of scientists, impacted people, software engineers, layperson, etc)
- describes what your demo will do by answering the relevant questions from the list below:
  - what will a person learn by reading/ using your demo?
  - if it's interactive what will vary? what will be the inputs?
  - what specific result will you replicate?

The proposal will be graded on specification and may be resubmitted until successful.

The demo will be graded on specification and can be revised and resubmitted one time.

Project (50%)

The final project is a chance to dive deeply into one of the course topics

Date	Milestone	Submission format
2021-02-19	Area Selection	Consultation meeting and general questions
2021-03-01	Topic Selection	Objectives and scope of work
2021-03-15	Proposal	Problem statement, lit review, method
2021-04-02	Checkin	Consultation meeting and prelim result
2021-04-13	Rough draft	Draft ready for peers to read
2021-04-19,21	Presentation	talk in class
2021-04-26	revision plan	plan for final revision, minor extensions
2021-05-07	final paper	final paper submitted for grading

Class 1: Introductions

Introductions & Goals

Course Admin

- Brightspace
- Zoom
- Google docs or markdown in the future?
- Website

Learning outcomes

knowledge research

- identify common problems and solutions in scientific application of ML
- identify common challenges and solutions for social applications: fairness,
- implement and extend research papers

Activities

- reading and evaluating ML research papers
- facilitating and participating in class discussions of the papers
- producing a replication, demo, or illustration of one concept covered for a broader audience
- completing a project using ML in a scientific or social domain
- reflect on methodologies used in this type of research
- writing a CS conference style (short & concise) final paper on their project

Model Based ML and this course

https://www.mbmlbook.com/toc.html

- missing data
- noisy or missing labels
- multiple objectives

We will look at a range of strategies for identifying and mitigating these problems including:

- robust evaluation
- model inspection
- explanations
- interpretable models

ML and Probability Review

admin

- collaborative notes
- brightspace will be updated later this week
- grading details by Wed
- environment for coding demos

More formalism

- model
- prediction algo
- cost function
- objective

Probability

- sample distros

Practical Application of ML & Pipelines

Class 3: ML Pipelines

Goals when using ML

1. Understand about the data (data science/ actual science) probability more statistics, maybe fit another examine model parameters, inspect them
2. understanding about Naive bayes fit different data varies
3. claims about the learning algorithm run multiple algorithms on the same data possibly multiple data

Basic setup

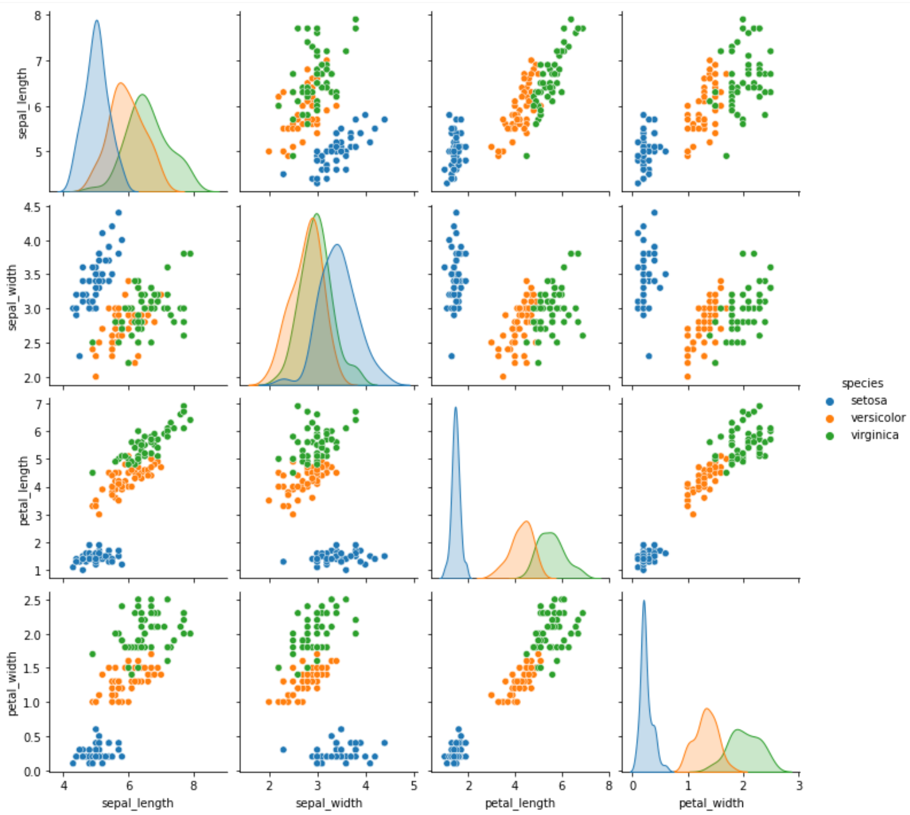
1. test train
2. training parameters
3. estimator objects
4. fit model parameters
5. metrics
6. cross validation

```
import pandas as pd
import seaborn as sns
import numpy as np
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import confusion_matrix, classification_report
from sklearn import datasets
```

```
iris_df = sns.load_dataset('iris')
```

```
sns.pairplot(iris_df, hue='species')
```

```
<seaborn.axisgrid.PairGrid at 0x7f4a66abbc10>
```




```
X,y = datasets.load_iris(return_X_y=True)
```

```
X.shape
```

```
(150, 4)
```

```
y.shape
```

```
(150,)
```

```
X_train, X_test, y_train, y_test = train_test_split(X,y,)
```

```python
gnb = GaussianNB()
```

```python
gnb.__dict__
```

```
{'priors': None, 'var_smoothing': 1e-09}
```

```python
gnb.fit(X_train,y_train)
```

```
GaussianNB()
```

```python
gnb.__dict__
```

```
{'priors': None,
 'var_smoothing': 1e-09,
 'n_features_in_': 4,
 'epsilon_': 3.1814253826530606e-09,
 'classes_': array([0, 1, 2]),
 'theta_': array([[4.97368421, 3.40789474, 1.46052632, 0.24473684],
        [5.92631579, 2.78421053, 4.26842105, 1.33684211],
        [6.66388889, 2.98888889, 5.61388889, 2.03055556]]),
 'sigma_': array([[0.1166759 , 0.14283241, 0.02975762, 0.01141967],
        [0.25246538, 0.0765928 , 0.22795014, 0.03916898],
        [0.42508488, 0.11876544, 0.33508488, 0.07545525]]),
 'class_count_': array([38., 38., 36.]),
 'class_prior_': array([0.33928571, 0.33928571, 0.32142857])}
```

```python
X_test[0]
```

```
array([6.2, 2.2, 4.5, 1.5])
```

```python
y_pred = gnb.predict(X_test)
```

```python
y_pred[:5]
```

```
array([1, 2, 2, 2, 1])
```

```python
y_test[:5]
```

```
array([1, 2, 2, 2, 1])
```

```python
confusion_matrix(y_test, y_pred)
```

```
array([[12,  0,  0],
       [ 0, 12,  0],
       [ 0,  1, 13]])
```

```python
gnb.score(X_test,y_test)
```

```
0.9736842105263158
```

```python
gnb2 = GaussianNB(priors=[.5,.25,.25])

gnb2_cv_scores = cross_val_score(gnb2,X_train,y_train)
```

```python
np.mean(gnb2_cv_scores)
```

```
0.9458498023715414
```

```python
gnb_cv_scores = cross_val_score(gnb,X_train,y_train)
```

```python
np.mean(gnb_cv_scores)
```

```
0.9458498023715414
```

```python
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        12
           1       0.92      1.00      0.96        12
           2       1.00      0.93      0.96        14

    accuracy                           0.97        38
   macro avg       0.97      0.98      0.97        38
weighted avg       0.98      0.97      0.97        38
```

```python
gnb.predict_proba(X_test)
```

```
array([[3.76605232e-101, 9.91921035e-001, 8.07896474e-003],
       [1.99591649e-201, 4.24899006e-006, 9.99995751e-001],
       [9.11299952e-145, 3.23334642e-002, 9.67666536e-001],
       [1.19176259e-217, 2.40355547e-008, 9.99999976e-001],
       [5.18022183e-063, 9.99985660e-001, 1.43399117e-005],
       [8.49651881e-055, 9.99995139e-001, 4.86068760e-006],
       [1.00000000e+000, 4.27476224e-016, 1.96497683e-023],
       [1.00000000e+000, 8.91604563e-021, 6.45539664e-025],
       [3.93683093e-152, 6.17418703e-001, 3.82581297e-001],
       [3.51192794e-185, 1.01327840e-005, 9.99989867e-001],
       [1.00000000e+000, 1.61789435e-017, 5.12910077e-024],
       [1.00000000e+000, 1.28683966e-018, 2.00275095e-025],
       [1.58971202e-040, 9.99998850e-001, 1.15027262e-006],
       [1.38152250e-149, 5.45453133e-002, 9.45454687e-001],
       [1.00000000e+000, 7.10196151e-018, 1.14509190e-024],
       [3.28289721e-226, 2.26510395e-006, 9.99997735e-001],
       [2.67031240e-217, 3.47277220e-007, 9.99999653e-001],
       [1.00000000e+000, 1.80600521e-019, 4.14917686e-024],
       [1.12016692e-129, 3.29647713e-001, 6.70352287e-001],
       [1.57237976e-096, 9.98754433e-001, 1.24556749e-003],
       [3.33965023e-077, 9.99793177e-001, 2.06823220e-004],
       [3.60215265e-111, 8.36788795e-001, 1.63211205e-001],
       [1.65590514e-113, 6.87056714e-001, 3.12943286e-001],
       [3.83392122e-232, 3.57008377e-010, 1.00000000e+000],
       [4.03452835e-055, 9.99977716e-001, 2.22835446e-005],
       [1.00000000e+000, 5.69515166e-016, 4.33059173e-022],
       [1.00000000e+000, 4.64788320e-013, 7.75019274e-019],
       [1.00000000e+000, 6.41249731e-017, 6.03859100e-024],
       [1.00000000e+000, 6.99394656e-018, 4.80638790e-024],
       [1.28724479e-150, 2.42537722e-002, 9.75746228e-001],
       [1.05583162e-069, 9.99954735e-001, 4.52653360e-005],
       [6.97611742e-191, 1.01860982e-006, 9.99998981e-001],
       [6.68978082e-159, 7.18772647e-004, 9.99281227e-001],
       [1.00000000e+000, 1.10512182e-017, 5.95010814e-024],
       [1.85886857e-143, 1.11750238e-001, 8.88249762e-001],
       [4.48976407e-083, 9.99252393e-001, 7.47607059e-004],
       [1.00000000e+000, 2.49963212e-014, 3.85898287e-021],
       [4.59590295e-108, 7.99737476e-001, 2.00262524e-001]])
```

Class 4: Missing Data: Basic techniques

Evaluation of missing data at training

- mulitple imputation
- ML based was better than imputation which is better than dropping samples
- example datasets: 45% of patients have at least 1 missing value

Imputation

- Mean imputation:
  - insert the mean based onthe other values
- Hot deck
  - mean-like with similarity

- Multiple inputation
  - 3 diff ways

Imputation ML

- MLP
  - fully connected
- Self organization
  - competitive learning
  - NN on modle of nodes in 2d grid,
- KNN
  - select closest complete case to impute values from
  - expensive for large datasets due to need to search everywhere for each missing value

Testing

- Train NN based on data imputed with each technie

Conclusions:

- in general, any imputation was better than deletion
- ML based performed better

Discussion & Questions

- interesting that even simple methods provide improvement
- SOM is sort of unclear how does that work?
- Review of MLP and [sigmoid]

Handling missing values At application time

- reduced models vs imputation.
- broad approach
- 15 common datasets

Techniques:

- Discard
- Acquire missing values
- Imputation
  - predictive value imputation
  - distribution based
  - unique values
- Reduced Feature Models
  - retrain for different feature models

Feature imputability impacts the distribution or predictive type of imputation

More complex model

- decision tree with bagging
- again, reduced model is the best strategy

Hybrid Models for efficient prediction

- reduced models
- a hybrid is a complete model with stored subset for most common missing features
- Reduced feature enseble
  - N models for N features
  - each one is missing one feature
  - average these together for final prediction

    - substantial reduction in when there is a single feature is missing
    - combine with imputation for multiple features
    - relative accuracy is better than imputation

General takeaways

- reduced models vs imputatation is a large improvement
- this is sort of an imputation

Weaknesses

- Didn't check unique value imputation
- MCAR
- focused on

Overall Discussion

- How might the two problems interact?
    - if missing data at both train and prediction…
    - train using missing data without imputation for training the separate models
- Questions on these ideas
- What additional things might you need to consider when choosing one?
    - feature imputability at training
- what to do with time series data
- How to check if missing CAR?
    - look at collection technique
    -
- what do to with varying data per person
    - LSTM for time series data
    - hierarchichal modeling other wise
    - example of hierarchical with time series also

For Wednesday

1. Graphical Models for Inference with Missing Data
2. Missing Data as a Causal and Probabilistic Problem

Missing Data 3

Handling Missing Data in Decision: A probabilistic approach

key ideas

- A decision tree's structure and notation
- Review of imputation
    - Predictive value imputation
        - mean, median or mode
        - make assumption that features are indpendent
        - surrogate splits, partition data using another feature to
- XG Boost

Expected Predictions:

- impute all possible completions as once to avoid strong dist assumptions
- consistent for MCAR and MAR
- expensive, but density can help reduce
- tractably compute the exact expected predictions
- loss minimization

Experiments

- for a single dataset, outperforms in general

Discussion

- generally easier
- given single dataset, of results, how much do we trust this?
- what does this provide as an advantage
- NP hard

How to miss data?: Reinformcent leanring for environgments iwith high obseration cost

Key points

Reinforcement learning

- cost associated with making accurate observations
- goal directed
- RL agent tries to

Problem setting:

- $o \sim p_0(o_t |s_t; \beta)$
- beta is accuracy og obs
- r is old reward

Scenario A:

- observed cangle vs

Big picture: manipulating how the data collection

Discussion

- survivorship bias?
- right left imbalance for figure 3
- simple pendulum example helped overcome the background lacking
- figures

General

Try writing out a missingness graph for a problem of choice, some scenario where you imagine there would be missing data, or an example dataset that you can find.

MIssing data

supervised

Background

- Hadamard
-

Readings for next week:

http://sorelle.friedler.net/papers/fairness_comparison_fat19.pdf https://fairmlbook.org/

- introduction and classification chpaters (1 and 2)

Intro to Fairness

Comparison of Fairness Interventions

Paper discussion

By Sarah M Brown
© Copyright .