This week's discussions focused on GDPR and how legislation has affected the topic of interpretable machine learning. It also focused on different types of interpretable ML models Our discussion focused largely on specific articles of GDPR which outline requirements for non-discrimination, and for people to have access to data collected about themselves. After learning about the actual text of GDPR, the discussion turned to what the implications were, and how effective such legislation could be. Two interpretations were presented, each one offering a different view of GDPR's effectiveness at regulating interpretability in machine learning. Finally, the class was given an overview of both post-hoc interpretable models and intrinsically interpretable models. The details of each of these models is provided in the notes.

Discussed last week

- Need for interpretability
    - What humans need interpretability for
    - Trust Causality Transferability, etc...
- Characteristics of Interpretability

GDPR- What is it?

- Restricts automated decision-making models that have a significant impact on users
- Ensures that the privacy of your data is protected
- The right of citizens to receive explanations for any automated decisions that impact him/her
- Any decision made by automated algorithms should be contestable in a court of law
    - Decision should be reversible if innocence is proven

GDPR

- When does it go into effect?
    - April 2018
    - Affects all EU members
- What is it replacing
    - Data protection Directive (1995) in the EU
- Differences between GDPR and DPD
    - GDPR is an actual law, DPD was a directive
    - Penalties/punishments that GDPR imposes
        - 20 million euros or 4% of global revenue (whichever is greater)
    - It now covers all EU citizens, including those residing outside the EU
- Article 22- Prohibits any automated decision making unless conditions are met

Non-Discrimination

- Algorithmic profiling is discriminatory
- Paragraph 71-What is it?
    - Explicitly requires data processors to ensure that no discriminatory effects are brought upon data subjects on the basis of processing sensitive data
- Paragraph 71 and Article 22 Paragraph 4
    - Specifically address discrimination from profiling that makes use of sensitive data

Two Interpretations of Paragraph 71

- Minimal Interpretation
    - Just get rid of protected features
    - Pros: Simple to implement
    - Cons: You don't address correlations between protected and unprotected features
- Maximal interpretation
    - Get rid of all features that might be correlated with protected features
    - Pros: you get a legal model
    - Cons: you might lose predictive accuracy
    - Cons: you might run into the problem of uncertainty bias

What is uncertainty bias?

- One group is under-represented in the sample, so there is more uncertainty associated with predictions about that group
- The algorithm is risk averse, so it will ceteris paribus prefer to make decisions based on predictions about which they are more confident

The Uncertainty bias experiment

- Two population groups, whites and non-whites
- Algorithm used to decide whether to extend a loan, based on predicted probability that the individual will repay the loan
- Generated 500 synthetic people, and vary the number of non-whites
- True probability of repayment/non-crime set to 95%
- Check whether lower end of the 95% confidence interval is above a fixed approval threshold of 90%.
    - If yes, grant loan, else don't

Article 13-15

- Article 13
    - Data subjects have right to access information collected about them
        - Example: Facebook dump data- are they sharing all the data they are collecting from us? We don't know.
- Article 14
    - They should be notified about the collected data
    - Example: Android app notification for collection of location information
- When profiling takes place, right to meaningful information about the logic, or an explanation
- How to provide this transparency?

Critical Analysis of Famous Interpretable ML Methods

- Post-hoc interpretable methods
    - Treat ML model as a black box

- Design an explanation system which can infer what the black box might have learnt without looking at internals of ML model
- Partial Dependence Plot (PDP)
  - Pick a feature, vary the feature values
  - See what is the impact of changing those feature values
  - **As you change these values, how does predictive accuracy change?**
  - X-axis represents your manipulation of the values, y-axis represents predictive accuracy
  - Drawbacks
    - Looking at change over the entire dataset, average change of predictive accuracy
    - Does not show how data changes at different points in the dataset
  - Pros:
    - Intuitive
    - Easy to implement
  - Cons:
    - Assumption of independence
- Individual Conditional Expectation (ICE)
  - Instead of looking at average of all data, you look at **single data points**
  - Each line measures how changing of a feature value affects that single data point
  - Can have hundreds of different lines
  - Pros:
    - Intuitive
    - Easy to implement
  - Cons:
    - Assumption of independence
- Permutated Feature Importance
  - Does changing a feature change the overall predictive accuracy?
  - Shows the "feature importance", measuring how a single feature can affect the dataset
  - Pros:
    - Provide a highly compressed, global insight
    - Comparable across platforms
  - Cons:
    - Assumption of independence
- Global Surrogate
  - Train black box model to make a prediction
  - Train an interpretable model on the predictions from the black box model
    - I.e. training a linear model on data from a NN
  - Pros:
    - Any interpretable model can be used
    - Decision tree, linear model, etc.
  - Cons:

        - Can only explain the model, not the data
        - Surrogates might not be accurate
    - Local Surrogate
        - A model which mimics decision of a complicated model for one specific region of data
        - Will not be effective for all regions of the data, only one
        - LIME
            - Image on the left is divided into interpretable components
            - Generates a dataset of perturbed instances by turning some of the interpretable components "off"
                - Perturbed instances are close to the original image, so they lie in the vicinity of the original data point
            - For each perturbed instance, one can use the complicated un-interpretable trained model to get the probability that a tree frog is in the model
        - Pros:
            - Model-agnostic
            - Short, contrastive, human-friendly explanations
        - Cons:
            - Need to define a kernel to define the area in which data points are considered
                - Often an open problem, no good solution
    - SHAP (Shapley Value)
        - Prediction can be explained by assuming that each feature value of the instance is a player in a game
        - Different from surrogates because it is not learning a new ML model
            - Rather it is similar to PDP, ICE, FP because it is trying to determien feature value
        - Contribution from each player is measured by adding and removing the player from all subsets of the rest of the players
        - The Shapley value for one player is the weighted sum of all its contributions
        - **Current state-of-the-art, should be used today**
        - Cons:
            - Very complex, computationally expensive
    - Some quick points to consider
        - Do you need to understand the whole logic of the model, or do you only care about the reasons for a specific decision?
        - What is your time limitation? If the user needs to quickly take a decision it may be preferable to have an explanation that is simple to understand
        - What is the user's expertise level? Do they understand ML well, not at all, a little?

Intrinsically Interpretable models

- Decision Trees- How do they work?

    - Pros:
        - Graphical structure
        - Work with subset of attributes
        - Hierarchical representation dictates the order features are considered in
    - Cons:
        - Irrelevant attribute splits
        - Data fragmentation due to irrelevant splits
- Classification Rules
    - List of if-then-else logical statements
    - Pros
        - More interpretable
        - Easier to understand modular pieces
        - Enable aggressive pruning/limit number of conditions that are present inside these rules
            - Irrelevant splits can be removed
    - Cons:
        - No ordering of clauses, how to figure out which ones are more important?
        - How to measure feature importance? Average number of rule conditions tested
- Decision tables
    - Tabular representation of data
    - Pro: fairly interpretable, literally a table
    - Con: size of these tables can grow prohibitively long
- Nearest-neighbors
    - For any data point, look at nearby data points, whichever data point has majority, then the selected data point will match
    - Pros:
        - Can be improved using prototype points, which are single points representing one class category
    - Cons:
        - Explanation is different for every data point – each data point has a different set of nearest neighbors, hence a different explanation
- Bayesian Network Classifiers
- How to improve?
    - Don't use model size as a single measure of interpretability
    - Introduce semantic monotonicity constraints in model building
        - Probability of someone having cancer is going to go up as the age of a person goes up