CSC461 INTRODUCTION TO DATA SCIENCE

Dr. Muhammad Sharjeel

https://muhmmadsharjeel.github.io/

11 DEBUGGING DATA SCIENCE

DEBUGGING DATA SCIENCE PROBLEMS

- Imagine to go about defining a data science problem, define input/output pairs for a prediction task, only to find out that when it is executed (doing prediction using the model), it doesn't work
- "What differentiates experts in data science from others is not what you do first, it's what you do second when that first thing doesn't work." - Kolter's Law

DEBUGGING DATA SCIENCE PROBLEMS

- Traditional debugging of programs is relatively straightforward
  - Beforehand knowledge of desired input/output pairs
  - Clarity of how each step in the algorithm (should) work
  - Tracing step-by-step the execution of the program (either through a debugger or with print statement) to see where the state diverges from the actual execution or to discover what and where is something wrong
- Data science debugging
  - Known desired input/output pairs
  - Clarity that a data science algorithm (should) work?
  - What can be traced to see why it may not be working?
- Debugging web scraping, etc, is much more like traditional debugging, however, the focus is largely on debugging data science prediction tasks
- In data science, the concept is "**debugging the problem**" instead of debugging the algorithm (model)

DEBUGGING DATA SCIENCE PROBLEMS

- Example: We want to predict whether in a large factory any given machine will fail within the next 90 days
    - We are given signals monitoring the state of a machine
    - Prediction task: binary response of whether the machine will fail

DEBUGGING DATA SCIENCE PROBLEMS

- Step 1: Determine if the problem is impossible
- Step 2a: What to do about impossible problems?
- Step 2b: What to do about feasible problems?
- Step 1b: Impossibly good performance

DEBUGGING DATA SCIENCE PROBLEMS

- Step 1: Determine if the problem is impossible
- Plenty of tasks that would be really nice to be able to predict
- However, absolutely no evidence that there are necessary signals (available data) to predict them
  - Example: predicting stock market from Twitter
- But, hope springs eternal, and it’s hard to prove a negative

- *Step 1: determine if the problem is impossible*
- **Figure out if the problem could be solved manually**
  - Create an interface where a human can play the role of a prediction algorithm
    - Manually make predictions of the outputs given the available inputs
- Need some intuitive way of visualizing what a complete set of input features looks like
  - Tabular data for a few features, raw images, raw text, etc.
- Similar to a data science algorithm, a human can refer to training data (known labels), but can’t peak at the answer on test/validation set

DEBUGGING DATA SCIENCE PROBLEMS

- It’s a common misconception that machine learning will outperform human experts on most tasks
- In reality, the benefit from machine learning often doesn’t come from superhuman performance in most cases, it comes from the ability to scale up expert-level performance extremely quickly
- If a human can’t make good predictions, neither will a machine learning algorithm

DEBUGGING DATA SCIENCE PROBLEMS

- Can the prediction problem be solved manually?
- Impossible problem
    - Go to Step 2a
- Feasible problem
    - Go to Step 2b

DEBUGGING DATA SCIENCE PROBLEMS

- Step 2a: What to do about impossible problems?
- We have built a classification model, run through many cases, however, got poor performance
- Do not try to throw more sophisticated, bigger, or complex data science algorithms at the problem
- Instead, try changing (different aspects of) the problem
    - Changing the input (i.e., the features)
    - Changing the output (i.e., the problem definition)

DEBUGGING DATA SCIENCE PROBLEMS

- Changing the input
- A fact: adding more features is what makes the problems “impossible” instead of impossible
    - Most of the time it’s a false hope that one “magical” feature will make the problem easy
- Adding more features (data) is good, but:
    - Always, spot check (visually) to see if the new feature(s) can help in differentiating between what was previously difficult to predict
    - Get advice from domain experts, see what sorts of data source they have use

DEBUGGING DATA SCIENCE PROBLEMS

- Changing the output
- Just make the problem easier!
- Instead of trying to predict the future, try to predict what an expert would predict given the available features
- Example: For predictive maintenance task, shift the question
    - Would this machine fail?
    - Would an expert choose to do maintenance on this machine?
- With this strategy we already have an existence proof that it's feasible

DEBUGGING DATA SCIENCE PROBLEMS

- Changing the output
- Move from a question of getting “good” prediction to a question of characterizing the uncertainty of the predictions
- Seems like a cop-out, but many tasks are inherently stochastic
- Try to quantify the likely uncertainty in output given the input
    - Example: If 10% of all machines fail within 90 days, it can still be really valuable to predict if whether a machine will fail with 30% probability

DEBUGGING DATA SCIENCE PROBLEMS

- Step 2b: What to do about feasible problems?
- Good news! the prediction problem seems to be solvable
- We have devised a solution for a data science problem, only to find out that it doesn't work
    - Performs much worse than a human
- Again, throwing more algorithms, data, features, etc. is unlikely to succeed
- Instead try building diagnostics that can check what the actual problem may be

DEBUGGING DATA SCIENCE PROBLEMS

- Consider the training and testing loss of the algorithm by plotting over different numbers of samples
- Determine if the problem is one of high bias or high variance
- For high bias, add features based upon your own intuition of how you solved the problem
- For high variance, add data or remove features (keeping features based upon your intuition)

- The problem may genuinely be “AI Hard”
  - It relatively easy for a human to solve it, but a computer cannot
- These are typically the most interesting problems from a research standpoint

DEBUGGING DATA SCIENCE PROBLEMS

- Step 1b: Impossibly good performance
- We have developed a data science solution right off the bat, and it works great!
- Be sceptical, unless it's "obvious" how to solve the problem
- Try to solve the problem manually
  - What if a human can't? be extremely sceptical
- Sometimes it's very easy to accidentally set up a problem that lets the algorithm "cheat"
  - Using a feature that is virtually a deterministic one-to-one function of desired output

DEBUGGING DATA SCIENCE PROBLEMS

- The confusion matrix to debug data science problems

	Model performs well	Model performs poorly
Human predict well	Congrats, but still be sceptical and do a bit of analysis	“Feasible” problem, debug your data science algorithm
Human predict poorly	Be sceptical, make sure is it not cheating	“Impossible” problem, debug the problem

THANKS