Mental Health Trends in Tech

Kaylynn Crawford
Machine learning analysis for SDS 296. Smith College 2016.
View here: https://silentttone.github.io/MentalHealth/

Introduction

My goal for this project was to identify policies that companies can implement that encourage good mental health policies. I found a dataset from a survey done by OSMI[1], a nonprofit. It has 59 questions on topics ranging from personal mental health, information about the respondent's company policies, and demographic information.

Using this information, my goal is to determine what a company can do in order to encourage employees with mental health issues to get treatment.

To do this, I built a classification tree with "Treatment" as a response, and figured out which factors best determined if a person with a mental health disorder was receiving treatment. The factors at the top of the resulting tree will be the ones that make the largest difference for employees' mental health, and therefore are the ones that companies should implement.

Data & Methods

I used Python and the sklearn library[2] to clean and analyze the data. First, I read in the CSV file I'd gotten from Kaggle[3], then I cleaned up the data by removing unnecessary rows and changing the types of the variables from object to category. Since my goal is to predict when people with mental health disorders get treatment, I also removed all the rows where the respondent does not have a mental disorder.

Next, I removed the columns where over 400 people did not respond to the question. Then, I removed any rows left with NaN values. At this point, I have a data set with 381 responses, all with a mental health disorder, and no NaN values.

Then I turned the categorical variables into binary variables using one-hot encoding[4] because Python's decision tree method only takes numerical values. One-hot encoding does not impose ordering on the data.

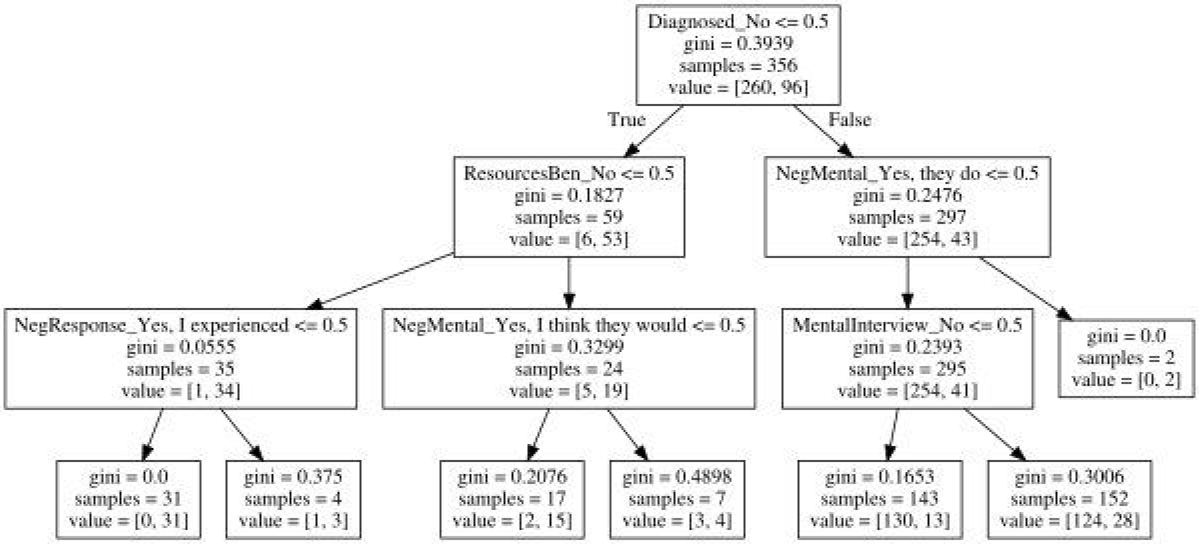
Lastly, I created decision trees with 1-6 levels, compared the accuracy and displayed the results.

Results

The best model was a three level tree. It correctly classified the most data without overfitting. When the data was split into a training and a test set, a six layer tree correctly classified 93% of observations, and had a 80% accuracy rate for determining whether an individual received

[1] OSMI: https://osmihelp.org/research/
[2] Sklearn library: http://scikit-learn.org/
[3] Dataset: https://www.kaggle.com/osmi/mental-health-in-tech-2016
[4] fmder's comment on March 29: https://gist.github.com/ramhiser/982ce339d5f8c9a769a0

treatment. A three level tree correctly classified only 91% of observations, but had an 84% success rate on the test data. This shows that a six level tree overfits the data, and the three level tree is a better model. The sample size is only about 350 observations, however, so both models would improve with more data.



The most determinative factor seems to be whether or not the individual is diagnosed, which makes sense. Of those not diagnosed, only 14% are receiving treatment, even though all said that they had a mental disorder.

Beyond that, the most important factors are whether the individual has resources to learn about mental health concerns and options for seeking help, and whether the individual believes that their coworkers would view them differently because of their mental health issues.

Conclusions & Future Work

Besides a diagnosis, the factors that are most important are having resources, and having supportive coworkers. For companies, the take aways from this should be to encourage employees not to judge others for any mental health disorders, and to provide resources in the form of both insurance and information.

The fact that the model can be relatively predictive, even with a low sample size, indicates that there are universal factors that can help encourage good mental health practices. Hopefully the survey by OSMI and research done with the resulting data helps to inform companies about good mental health policies.

If I had more time, I would definitely try the same analysis in R. The function that creates a decision tree in Python does not accept categorical variables, so the results were harder to interpret than I wanted.

I also think that the results would be more meaningful if there was a better way to deal with the fact that many questions had about five possible responses. I think this muddled the tree and made it harder to interpret.