A Survey of Mental Health in the Workplace

Austin Lutterbach [*1] **Kevin Jiang** [*1] **Haoran Zheng** [*1]

1. Background & Motivation

The 'crunch' environment, pressure to meet deadlines, and fear of failure are just some of the stressors that affect professionals in the Tech industry. Although issues of stress are by no means unique to the Tech industry, there has been an increasing interest in identifying and supporting stressed tech workers in recent years (e.g. Murphy et al. [1]).

However, despite such progress, many people who need help for a mental issue do not seek it. In fact, a study by the World Health Organization [2], found that between 30 and 80 percent of people with mental health issues do not seek treatment. If these people can be readily identified, health organizations or tech companies can more easily provide necessary resources for these people of interest. In other words, providing care to those suffering from mental-health related problems will undoubtedly prove useful not only to the operations of large-tech companies but, more importantly, to the well-being and health of workers.

Problem Statement & Weapon of Math Destruction

The objective of our data analysis research is to thus build a classification model that determines if a person of interest should seek treatment.

Our problem is inherently a *Weapon of Math Destruction*, as telling workers they should seek treatment based on our model findings may very well affect their decision to seek help or alter their self-image/perception. Therefore, in light of the possibility of reinforcing pre-existing conditions, it is important to handle and utilize our developed model with care.

2. Data Set Preprocessing

Description

The data set is from a 2014 survey launched by Open Sourcing Mental Illness (OSMI) [2] that measures attitudes towards mental health and frequency of mental health disorders in the tech workplace. With over 1200 responses, the data set covers over 25 variables, from demographic information to employment information. We believe the variables and data set are relevant enough to build an effective machine learning model as proposed in our objective.

The purpose of the project is to build a classification model which can recognize people who suffer from mental health issues but do not seek treatment. Before using primary and advanced algorithms to build models, data preprocessing will be conducted to get dependent and independent variables.

Feature Transformation

Several steps were taken to ensure that the data was appropriate to work with for our problem. For the independent variable, since no feature directly points out those who are suffering from mental issues but not seeking help, features treatment and workinterfere would be used to get the independent variable. The definition of treatment is that "Have you sought treatment for a mental health condition". And the definition of workinterfere is that "If you have a mental health issue, do you feel that it interferes with your work?". There are five types of value within the workinterfere column, including "NA", "Rarely", "Sometimes", "Often", "Never". Therefore, respondents who are not reporting "NA" are actually the ones with mental health conditions. Through filtering the "NA" records, treatment columns can be served as independent variables directly. For the dependent variables, features country, comments, state and timestamp would be dropped because of data scarcity and model simplicity. All the other features left would serve as potential dependent variables. After initial data processing, the dataset includes 994 records and 23 features in each record.

Since machine learning models require all input and output data to be numeric, data encoding should be conducted to make the categorical data in the original dataset become integer. Two types of encoders would be used: Label Encoder and One Hot Encoder. To begin with, we use Label Encoder to convert the categorical data into model-understandable numerical data. With the help of sklearn.preprocessing.LabelEncoder in Python, all the features except Age are mapped to integers. A sample of the label-encoded dataset which is ready for modeling is seen in Figure 1. Feature selection would be conducted on the label-encoded dataset. However, the problem of label-encoded dataset is that the model will misunderstand the data to be in some kind of order since there are different numbers in the same column. Therefore, one-hot encoder would be conducted on the selected feature dataset after feature selection. And the one-hot-encoded dataset would be used to build

classification models.

Figure 1. LabelEncoded data

	Age	Gender	self_employed	family_history	treatment	no_employees	remote_work	tech_company	benefits	care_options
0	37	0	0	0	1	4	0	1	2	1
1	44	1	0	0	0	5	0	0	0	0
2	32	1	0	0	0	4	0	1	1	0
3	31	1	0	1	1	2	0	1	1	2
4	31	1	0	0	0	1	1	1	2	0

Avoiding Underfitting / Overfitting

Due to the relatively small data set after the preprocessing step, we utilized cross validation, bootstrapping, and feature subselection in order to decrease the variance of our model and, consequently, reduce the problem of overfitting. Any underfitting is combated by developing more complex models, as is seen below.

Visualizations

Some preliminary visualizations were performed to highlight certain aspects of the data. The primary focus was to survey potential differences between gender as gender was suspected to play an important role in seeking treatment.

Figure 2. Differences in treatment seeking by gender

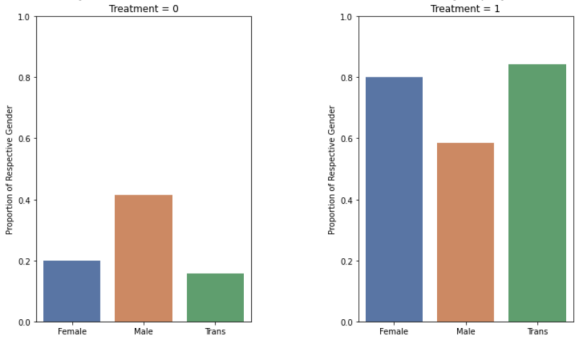


Figure 3. Differences in awareness of care options by gender

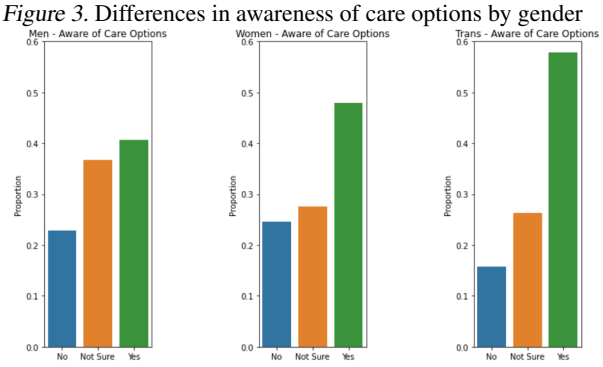
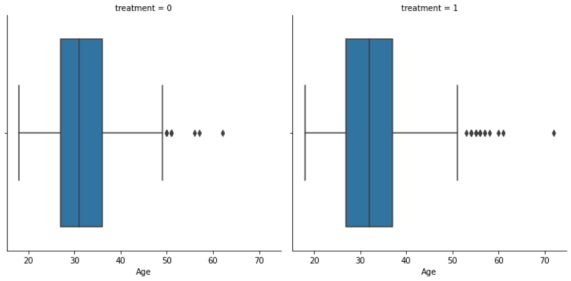


Figure 4. Box plot of age distribution between treatment levels



As we can see (Figure 2), there are slight differences among genders in treatment seeking as well as awareness of care options (Figure 3). Men are slightly more likely to not seek out treatment compared to women and trans individuals. Furthermore, men are more likely to be unsure about the potential care options they have at their company and are slightly less aware overall compared to women and trans individuals.

The distribution of age among treatment levels (Figure 4) are relatively similar, with those seeking treatment having a few more outliers as well as being slightly older than those who do not seek treatment.

3. Feature Subselection

Motivation

Before building prediction models, feature selection would be firstly conducted. The reasons are that feature selection enables the machine learning algorithm to train faster and reduces the complexity of a model and makes it easier to interpret. Also, it improves the accuracy of a model if the right subset is chosen as well as reduces overfitting.

Methodology

We use sensitivity and accuracy for classification models as our error metrics, because in our problem, we should penalize false negatives much more than false positives. Thus, we want a high sensitivity which reflects the model's ability to identify people who need treatment, and also a high accuracy so that the model does not trivially learn to predict every patient as needing treatment.

Furthermore, in all methods, we bootstrapped the original data set to obtain approximate confidence intervals for our results, so as to ensure any difference is statistically significant as opposed to occurring due to chance.

Finally, as a baseline we used a Random Forest Classifier to evaluate each sub-feature groups because it is a handy algorithm that can produce a relatively good prediction result even with the default hyper parameters. Also, overfitting

will not happen in random forest classifiers most of the time.

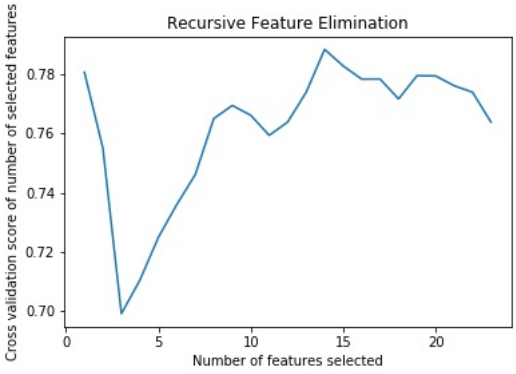
METHOD	SENSITIVITY	ACCURACY
ALL FEATURES	0.81± 0.021	0.71± 0.010
CORRELATION ANALYSIS	0.77± 0.016	0.70± 0.011
TREE-BASED	0.82±0.034	0.73±0.019
UNIVARIATE	0.80±0.025	0.72±0.020

Table 1: Sensitivities and accuracies for Feature Subselection Methods

RECURSIVE FEATURE ANALYSIS

We first use recursive feature elimination with cross validation to figure out how many features we need to achieve for best accuracy. This method is an effective approach for eliminating features from a training dataset and achieving automatic tuning of the number of features selected with cross validation. It is basically a backward selection of the predictors. The technique begins by building a model on the entire set of predictors and computing an important score for each predictor. The least important predictors are then removed, the model is re-built, and importance scores are computed again. This way, we specify the predictor subset's size that optimizes the performance criteria. As can be seen in Figure 5, the optimal number of features is 13.

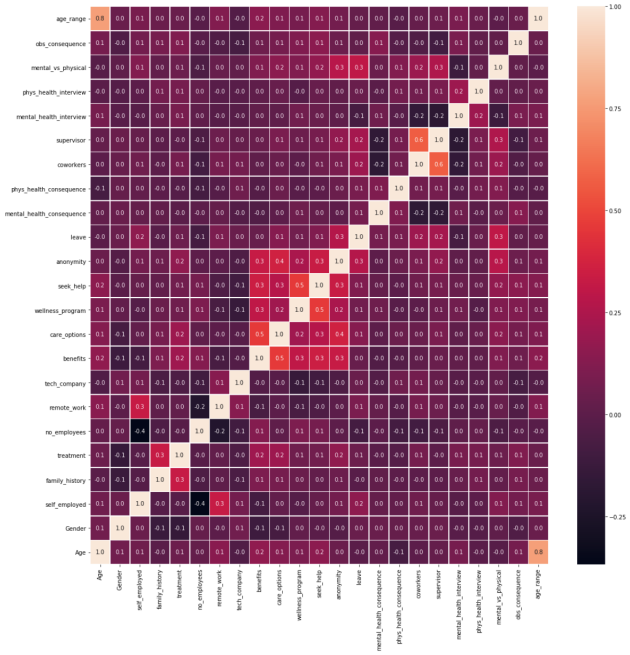
Figure 5. Recursive Feature Subselection



CORRELATION ANALYSIS FEATURE SELECTION

As can be seen in the Figure 6, the top 13 related features from correlation analysis are Feature A, including family history, benefits and etc. The classification accuracy from Feature A is 0.69. Furthermore, we see that we have moderate correlation between features like careoptions and benefits as well as coworkers as well as supervisors. There is a strong, obvious correlation between age range and age. Elsewhere, the features have insignificant correlation amongst each other.

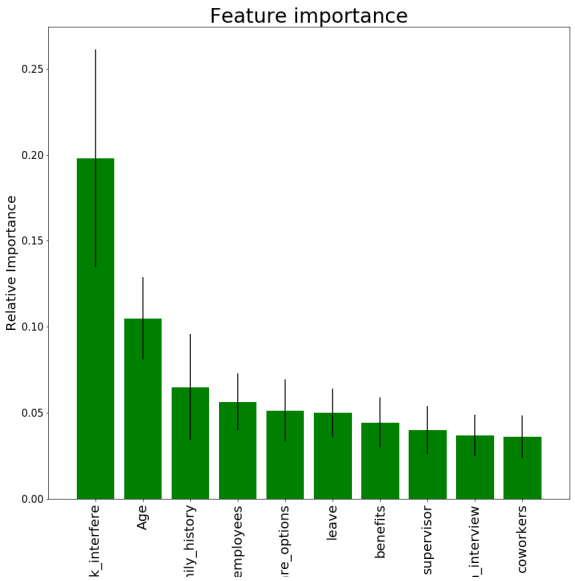
Figure 6. Correlation Matrix



TREE-BASED FEATURE SELECTION

As can be seen in Figure 7, the top 13 related features from tree-based feature selection are Feature B with a classification accuracy of 0.71.

Figure 7. Feature importance in tree-based feature selection



Univariate Feature Selection

The top 13 related features from univariate feature selection are Feature C and achieving a classification accuracy of 0.65.

Conclusion

Therefore, comparing the three selected feature subsets' performance, Feature B will be used to build more advanced prediction models. The comprehensive list of Feature B can be checked in the reference [4]. One-hot-encoded would be conducted on Feature B and the one-hot-encoded data is used to build our prediction models.

4. Model

Experiments

We tried the following models and tested for sensitivity and accuracy. Bootstrapping (see Figure 8) was performed to provide stronger estimates for the statistics in question.

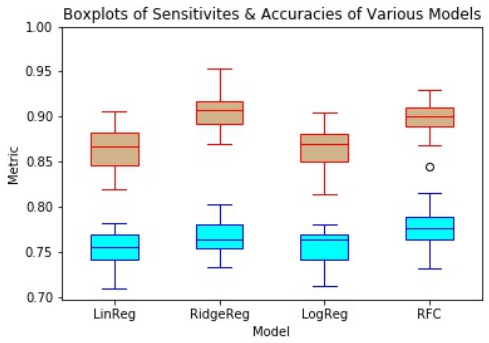
- Simple linear regression model with no regularization
- Ridge regression
- Logistic regression with regularization
- Random Forest Classification

The results can be seen in the following table:

Model	Sensitivity	Accuracy
Linear Regression	0.867 ± 0.025	0.755 ± 0.019
Ridge Regression	0.908 ± 0.021	0.763 ± 0.018
Logistic Regression	0.870 ± 0.022	0.763 ± 0.019
Random Forest	0.900 ± 0.020	0.777 ± 0.020

Table 2: Sensitivities and accuracies for varying models on bootstrapped data

Figure 8. Model results; top boxplots denote the models' sensitivity and the bottom boxplots denote the models' accuracy



Prediction

As previously stated, we decided upon using Random Forest Classification as the model to predict whether an individual would want to seek treatment based on their features. Using the features obtained from feature selection, the data was encoded using one-hot encoding (age was left untouched). The primary goal was to predict the proportion of individuals who seek treatment that belong to a specific feature and observe differences from the true population. For the scope of this project, we limited our predictions to gender, family history, and the overall population. To get a good estimate of the predicted proportions, bootstrapping was performed on the test data to more accurately reflect the statistic (proportion) of our predictions.

Gender

Our model was more likely to predict that women and trans individuals would seek treatment and that men were slightly less likely to seek treatment compared to the true population (Figure 9). Part of this stems from the fact women and trans individuals were underrepresented in the sample (women $n = 26$ and trans $n = 4$) compared to men ($n = 70$), thus analysis on these features are limited in scope and completely infeasible for trans individuals. Increasing the training set (10 percent to 20 percent) split as well as using ridge regression helps prevent these false positives (Figure 10) at the cost of accruing false negatives.

Figure 9. RFC Prediction: Proportion of individuals who seek treatment by gender

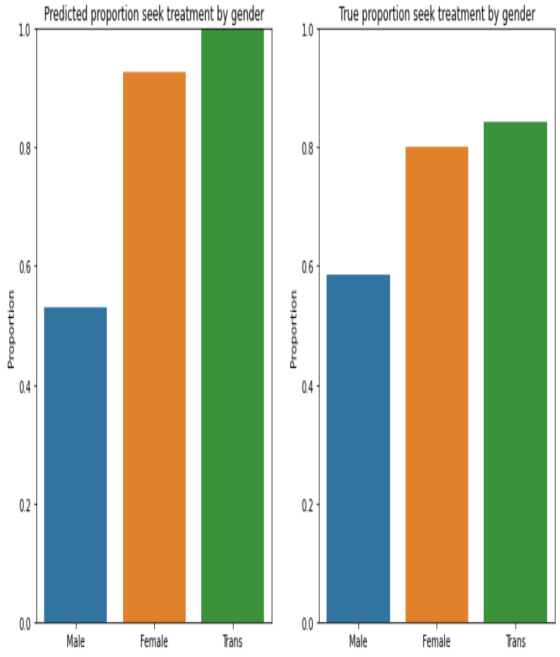
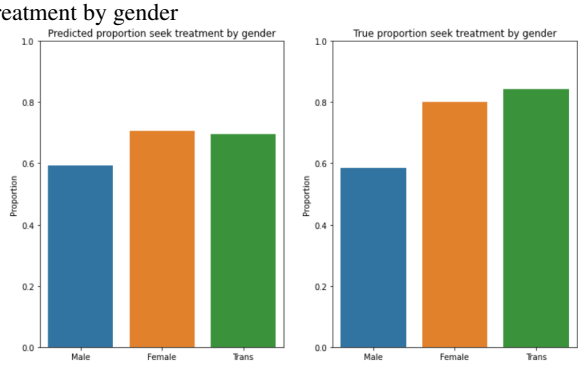


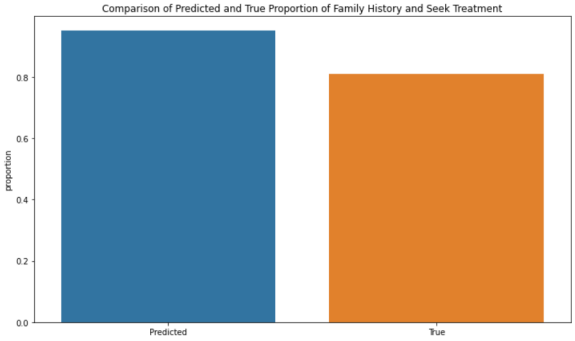
Figure 10. Ridge Prediction: Proportion of individuals who seek treatment by gender



Family History

Our model had more false positives for individuals who had a family history of mental illness compared to the true population (0.93 vs 0.81). This may be caused again by a relatively small sample size ($n = 41$) which skews towards more false positives. Increasing the training split increases the sample size to ($n = 91$) but only decreases false positives by one percent, suggesting our model may be biased towards family history.

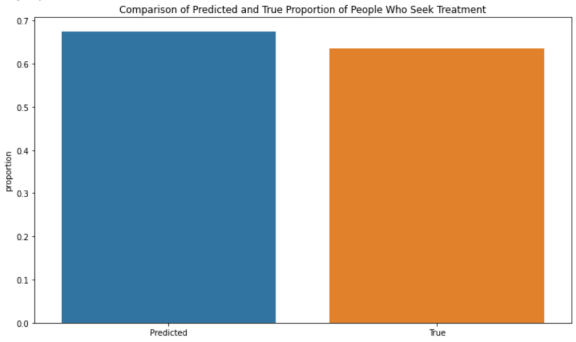
Figure 11. Prediction: Proportion of individuals who seek treatment by family history



Population

Overall, our model was reasonably close to the true population in predicting individuals who sought treatment (0.68 vs 0.64). Again, increasing our training split to 20 percent decreased the false positive rate to right on mark with the true population (0.64 vs 0.64).

Figure 12. Prediction: Proportion of individuals who seek treatment



5. Survey with Generalized Low Rank Models (GLRMs)

Note: Although we have only conducted preliminary data analysis with this method, the methodology involving Generalized Low Rank Models (GLRMs) is worth noting for future work. Thus, this section can be seen as an algorithmic development and discussion centering around GLRMs.

Motivation

GLRMs give low-rank representations of observed data A by capturing its most important information through a learned representation X of the data and a weight matrix W. The latter is a linear combination of features which serve as condensed features or archetypes, and can be used in recovering the original data matrix A given a learned low-rank representation of X.

Methodology

In this approach, we learn a $k-$dimensional low-rank representation $X \in \mathbb{R}^{n\times k}$ and a weight matrix of archetypes $W \in \mathbb{R}^{k\times d}$ by solving the Generalized Low Rank problem

$$\sum_{(i,j)\in\Omega} \ell_j[A_{(i,j)} - XW] + \lambda_1 \cdot r_1(X) + \lambda_2 \cdot r_2(W)$$

where $A \in \mathbb{R}^{n\times d}$ is the data matrix, Ω are the indices of observed entries, $r_i(\cdot), i = 1, 2$ are regularizers on the learned matrices, and ℓ_j is a loss function defined on each column/feature of data. $k, \lambda_i, i = 1, 2$ are typically chosen via cross validation.

In order to simulate learning from missing data, we removed 10% of features from the training data A_{train} (with the label y), ran alternating minimization via the GLRM package in Julia [5], and extracted the weight matrix W. We then repeated this procedure multiple times in order to, effectively, create a sort of modified bootstrap and ultimately averaged the learned weight matrix $\hat{W} = \sum_{i=1}^{\#\text{ bootstrap}} W_i$.

We then multiply this averaged weight matrix with the $k-$ dimensional low-rank representation of the test data (without the label y) in order to recover $\hat{X}_{test} \in \mathbb{R}^{n_{test} \times d}$ where n_{test} is the # of test data points and. more importantly, whose last column reflects the predicted label $\hat{y}$.

Figure 13. Illustration of methodology for attaining $\hat{W} = W_{avg.}$

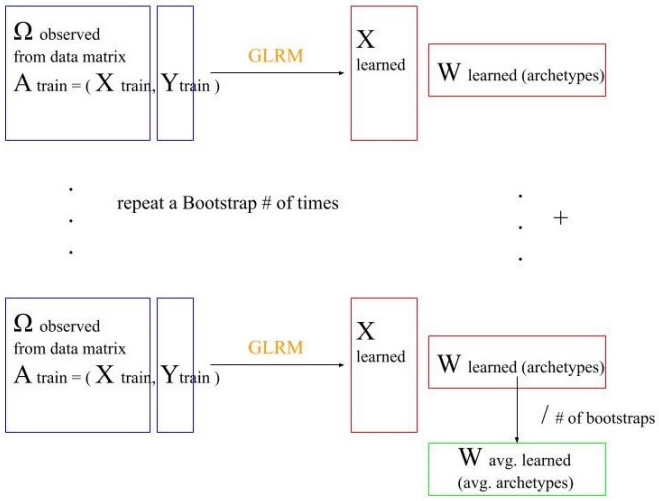
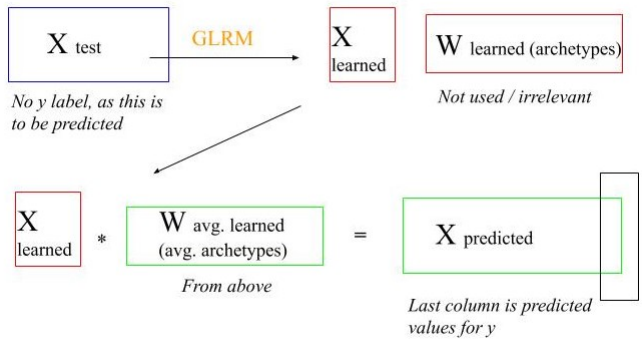


Figure 14. Illustration of methodology for attaining y_{pred}



Although this algorithm has the theoretical underpinnings as described above, when applied to our current data analysis project, we obtained subpar results and could not reach a performance that is on par with methods covered in previous methods. Nevertheless, such GLRM related algorithms can definitely be pursued in future work.

6. Current Limitations & Future Work

Limitations

Despite our best efforts to reduce variance and generate wherever possible via bootstrapping, the scope and size of the current OSMI data set used in this project is major limitation to any further analysis. For instance, any attempts to determine the effects of particular features on certain demographics is nearly impossible: only 4 individuals in the test data were reported to be transgender, so conclusive analysis of, say, the effects of *# of days with leave*, can not be conducted on this demographic. In a similar vein, the model tends to return a false positive when the person of interest is female or their family has a history of mental health issues – thus suggesting that our model may learn biases that are either inherently wrong or not present. Currently, women remain underrepresented in the tech industry [6], thus data in general is quite limiting for understanding these demographics. However, these limitations to our data could be remedied by obtaining more a comprehensive and larger data set.

Future Work

Aside from collecting a more extensive, thorough data set, there is much future work in developing sophisticated, state-of-the-art models: (i) Neural Networks or Deep Learning Models learn non linear functions of data through a cascade of linear models; (ii) GLRMs have proven to succinctly capture archetypes of various types of features (e.g. boolean, ordinal, real, etc.) and low-rank representations of the data; (iii) Explainable Boosting Machines [7], a form of Glass Box Learning, offer easier interpretability often without sacrificing a significant loss in important metrics such as sensitivity and accuracy.

Such models and methods, even like the GLRM-based algorithm proposed in Section 5, offer numerous opportunities for future work.

7. Conclusion

In this project report, we investigated the possibility of correctly identifying persons in the technology-related workplace who do or do not seek treatment for any existing mental health problems. After conducting preprocessing & feature-subselection, building several machine learning models, and proposing future directions of research based on GLRMs, we successfully constructed a model with sensitivity of ≈ 0.90 and accuracy of ≈ 0.78, a strong basis for classifying whether an individual seeks treatment. This model can be used to help identify those suffering from mental illness, but may not necessarily give the obvious signs such as significant work interference. Furthermore, our results can give companies an idea of individuals who

may need closer observation on their habits and to potentially recommend or make aware of treatment options for mental health conditions.

However, any of our constructed models should be used and interpreted with caution, given the limitations of both the scope of the data set and the classification problem's inherent risk of being a Weapon of Math Destruction.

There is undoubtedly future work to be done in this research problem: from collecting a more comprehensive data set to constructing more modern, state-of-the-art models, there are many possible venues to explore and expand upon from the methods employed in our data analysis project.

8. References

[1] Christian Murphy and Jennifer Akullian. 2018. We're All in This Together: CS Students, the Tech Industry, and Mental Health (Abstract Only). In Proceedings of the 49th ACM Technical Symposium on Computer Science Education (SIGCSE '18). Association for Computing Machinery, New York, NY, USA, 1071.

[2] Open Sourcing Mental Illness (OSMI) https://osmihelp.org/research

[3] Feature C: 'Age', 'Gender', 'selfemployed', 'familyhistory', 'noemployees','remotework', 'techcompany', 'benefits', 'careoptions', 'wellnessprogram', 'seekhelp', 'anonymity', 'leave', 'mentalhealthconsequence', 'physhealthconsequence', 'coworkers', 'supervisor', 'mentalhealthinterview', 'physhealthinterview', 'mentalvsphysical', 'obsconsequence', 'agerange'].

[4] Feature B: Age familyhistory noemployees leave careoptions supervisor coworkers physhealthinterview benefits mentalvsphysical mentalhealthconsequence physhealthconsequence seekhelp Gender wellnessprogram anonymity agerange

[5] Udell, Madeleine, "LowRankModels.jl", https://github.com/madeleineudell/LowRankModels.jl

[6] Daley, Sam. "Women in Tech Statistics for 2020 (and How We Can Do Better)." Built In, builtin.com/women-tech/women-in-tech-workplace-statistics

[7] Nori, Harsha, et al. "Interpretml: A unified framework for machine learning interpretability." arXiv preprint arXiv:1909.09223 (2019).