Analysis and Prediction of Employee Satisfaction in the US

Saahil Jain, Anil Yildiz, Tim Yngesjö {sj2675, yildiz, timyn286}@stanford.edu

MOTIVATION & PROBLEM DEF.

Our goal is to predict and analyze the drivers of employee satisfaction in the US. Our problem statement is: can we predict a company's employee satisfaction rating (based on Glassdoor) without looking at any employee feedback (aka only using public information about the company itself)? In addition to shedding light on what company attributes are desirable for employees, such a model could be used to better evaluate potential employers when existing employee reviews are sparse. Given the amount of time we spend working, it is of utmost importance that employees are satisfied at their respective companies.

DATA ACQUISITION

Collection of Data: We gathered the majority of our data by scraping the following wbesites - **Glassdoor, Yahoo Finance, Google Trends, Web-Stat, LinkedIn** and partially from **Kaggle –** for company data.

Shrinking dataset size: For every new data source we included, the number of examples reduced due to either lack of data or ambiguity when pairing old to new data (see table below). For example, there are only ~4,000 US companies listed on the stock market (WSJ).

Our dataset: Our dataset before preprocessing consists of a total of 52 features where 5 of those are categorical features and the rest are continuous features. The label of a company example is the Glassdoor score of that company.

Data source	# examples after merge	# added features
Glassdoor data	85 000	4
Kaggle dataset	34 000	7
Yahoo Finance data	3074	33
Google Trends data	3074	5
Web-Stat data	3074	3
LinkedIn data	3074	1

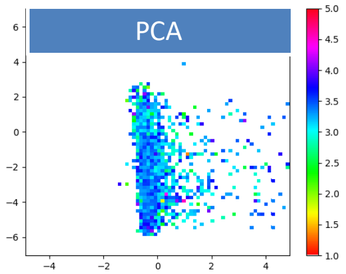
PREPROCESSING APPROACHES

Data cleaning: Cleaning the data involved handling missing values, converting strings to values, etc.

Feature selection: To select features and reduce noise, we used PCA and mutal information before running various models.

Categorical Encoding: To handle categorical variables like industry, we used the following two encoding scheme to turn them into 0 and 1 values: **One-Hot** and **Binary encoding.** Binary encoding increased the efficiency of our models with a smaller # of features than one-hot.

Filtering data: Since small companies often have very few reviews on Glassdoor, their score becomes more noisy / random. We therefore decided to filter out all companies with less than 300 employees.



MODELING APPROACHES

We focused on two main approaches: **regression** and **classification**. Although the Glassdoor score is discrete, if we assume it comes from a continuous underlying variable, it is natural to use regression. We also attempted classification with the motivation that the scores were 10 classes between 0 and 5 with the modeling assumption that the difference between a 3.3 and 3.5 rating is trivial so discretizing to the nearest half number is valid.

Compare regression and classification

To compare the performance of our particular regression and classification models, we defined the accuracy of a regression model to be:

$$\text{Accuracy} = \frac{1}{n}\sum_{i=1}^{n} 1\left\{\left|h\left(x^{(i)}\right) - y^{(i)}\right| < 0.25\right\}$$

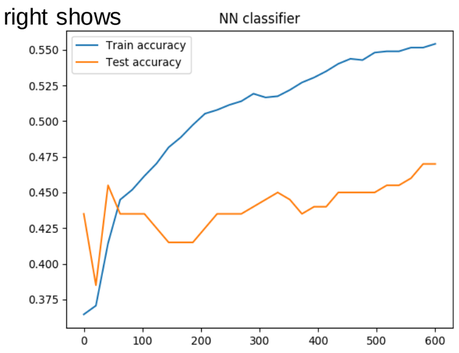
Methods Attempted:

Regression: Linear Classifier (L1, L2 reg.), Support Vector Regression, Neural Networks

Classification: Softmax, Support Vector Machine, Neural Network, K-NearestNeighbors, Decision Tree (w Adaboost), Linear Discriminant Analysis

RESULTS

As an example, the figure on the right shows the tradeoff between test and train accuracy per iterations for our neural network classifier.



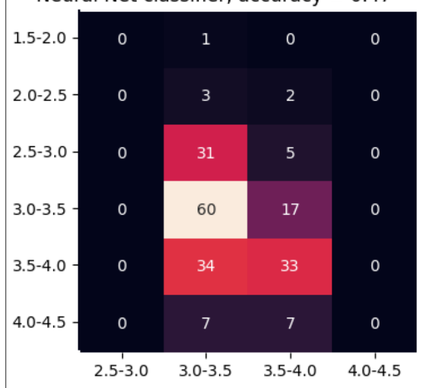
The following models in the table below show a range of approaches attempted with corresponding accuracies (for regression and classification) and mean squared errors (for regression).

Models	Train accuracy	Train MSE	Test accuracy	Test MSE
Support Vector Regression (C=0.9, rbf, eps=0.1, one hot, norm.)	0.76	0.09	0.5	0.18
Support Vector Regression (C=0.9, rbf, eps=0.1, binary, norm.)	0.71	0.11	0.49	0.187
Neural Net Classifier (fcn[70,20], binary, norm.)	0.55	N/A	0.47	N/A
KNN with PCA (neighbors=35, n_comp.=3, binary, norm.)	0.4	N/A	0.46	N/A
Linear Regression with L1 reg (alpha=0.005)	0.42	0.21	0.45	0.18
Neural Net regression (fcn[50, 30], binary, norm.)	0.53	0.15	0.44	0.28
Linear Regression with L2 reg (L2, alpha=0.001)	0.44	0.21	0.44	0.21
Softmax with PCA (C=0.8, n_components=11, binary, norm.)	0.4	N/A	0.43	N/A
SVM with PCA (C=1.5, rbf, n_components=6, binary, norm.)	0.51	N/A	0.42	N/A
AdaBoost (n_estimators = 10)	0.41	N/A	0.4	N/A
Softmax (C=0.8, binary encoding, normalization)	0.5	N/A	0.36	N/A
Decision Tree (max_depth=14, binary, norm.)	0.96	N/A	0.34	N/A
Decision Tree with PCA (max_depth=14, n_comp.=5, binary, norm.)	0.8	N/A	0.34	N/A

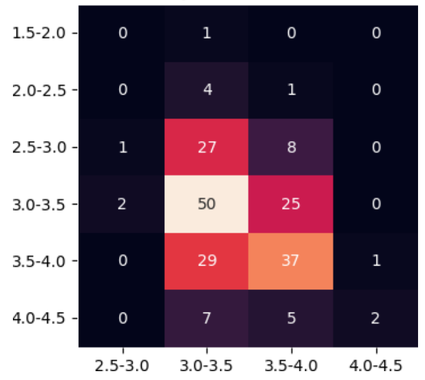
[1] Glassdoor,Ratings on Glassdoor, help.glassdoor.com/article/user/Ratings-on-Glassdoor, 2019.
[2] N. Thin, Social happiness: Theory into policy and practice. Policy Press, 2012.

The two images below show compressed heatmaps of the confusion matrices (row = true_y, column = pred_y) for neural net classifier (below left) and support vector regression (below right).

Neural Net classifier, accuracy = 0.47



SVR, accuracy = 0.485



ANALYSIS & CHALLENGES

Model Analysis

1. Support Vector Regression performed comparatively well as regression captures continuous nature of ratings and RBF kernel can efficiently map the input to a higher dimensional feature space for more complex analysis.
2. In general, PCA improved accuracy of classification models (KNN, softmax, and SVM), indicating initial feature set was noisy and could be compressed.
3. Binary encoding significantly improved performance of most models (except support vector regression by small margin). Compared to one hot encoding for categorical features, binary encoding reduced the # of extra features needed to capture a particular category, reducing overfitting / increasing speed.
4. Interestingly, neural net class. performed better than neural net regression.

Feature Analysis

1. **Mutual Information scores** for each feature were calculated. 5 most informative features for employee satisfaction are LinkedIn followers, total # of employees, stock 52 week high /low, and market cap.

2. **Coefficients for Regression** were inspected. The 3 highest weighted features were: profit margin, quarterly revenue share, and return on equity. The 3 lowest weighted features were: quarterly earnings growth, Google trends, and diluted earnings per share.

3. **Ablative Analysis** was performed. Most important features were state, city, country, industry, and return on assets.

Challenges: The biggest challenge related to the quantity / quality of our data. Structured public information about a company is limited / expensive and is weakly useful for predicting employee satisfaction. Additionally, Glassdoor scores are noisy, especially for smaller companies.

Future Work: Increasing the quantity of data (collecting private company info, adding international companies, synthetic approaches like data augmentation, etc.) as well as the quality of data (using unstructured text from company website / careers page, averaging scores from a variety of company rating sites, etc.) would decrease randomness of certain results.

[3] P. Hajek and K. Michalak, "Feature selection in corporate credit rating prediction,"Knowledge-Based Systems, vol. 51, pp. 72–84, 2013.
[4] J. W. Slocum and H. H. Hand, "Prediction of job success and employee satisfaction for executives and foremen.,"Training & Development Journal, 1971