Project Report

Amit Frechter, Michael Fishman, Keren Gruteke Klein.

[GitHub repo]

1. Introduction

Our project has developed a tool to estimate how long employees will stay with a company. This tool may be helpful for HR analytics and recruitment platforms because it helps predict when employees might leave, improving the planning for hiring and keeping employees. Given how quickly jobs and roles change today, our product allows businesses to forecast and prepare for these changes. By predicting how long employees will stay, companies can better manage their hiring, keep employees longer, and plan future roles accurately. Our goal with this project is to change how companies manage their employees, using data to guide more strategic decisions.

2. Data Collection and Integration

We chose to work with the original profiles dataset, and without the companies dataset since most of the relevant information from the companies dataset exists in the additional dataset we incorporate (e.g., company size, industry, etc.).

2.1 Additional Data

We incorporated company data from an existing dataset (α = 17, 20 columns). We consider an item a company. Each item contains information about one company's reviews, salaries (per role), ratings, etc. Not all the companies from the additional dataset exist in the LinkedIn profiles dataset, but some do, leaving us with enough data to work with.
Link to the dataset – Company Reviews (kaggle.com)

2.2 Incorporate Data Into the Project

Matching Company Names

At first, we tried various ways to join the two datasets based on company names. Matching company names is challenging due to variations in spelling, abbreviations, and legal suffixes. Before attempting any matching, we cleaned the company names by converting them to lowercase, removing punctuation, and eliminating common legal suffixes such as "LLC" and "inc." We then explored several methods to match the cleaned names: (1) Regular Join: A simple join on the cleaned names.(2) Levenshtein Distance: This approach is based on the edit distance between names, with a threshold set to 3. (3) Soundex: Using the phonetic algorithm Soundex. (4) Jaccard Similarity: Comparing sets of characters in the names with a 0.5 similarity threshold. (5) Fuzzy Matching: Using the "fuzzywuzzy" library's partial ratio for fuzzy matching and setting the threshold to 90%.

Considering efficiency and accuracy, we mainly examined fuzzy matching as our non-trivial method and utilized bucketing to perform a more efficient join. Unfortunately, the calculations were still too long, so we worked with regular join since it was fast enough and left us enough entries to work with (~1M).

Preprocessing

We had to apply many preprocessing steps before working with our data:

1. Modify the meta_industries mapping to fit the names in our additional dataset.
2. Process company rating information - convert dictionary columns to float columns by taking their average (for example, company ratings, happiness aspects, per-role ratings, etc.)
3. "Explode" the "experience" column to enable manipulation of its values.
4. Filter out profiles of people who lack experience - people with less than two companies in their experience since we're trying to predict company transitions (we can't learn anything from a person who only worked in a single company).
5. Filter out profiles that had inconsistencies (multiple "Present" end dates, overlapping role periods, etc.) or ambiguous role periods (such as "7 years" without start or end dates).
6. Fill exp_title with the last title from exp_positions in case it was null (these columns seemed to complete each other - when exp_positions wasn't null, exp_title was).
7. Extract a person's academic degrees from the "education" column.
8. Create a target column "exp_months" for the months a person worked in a certain company.
9. Merge different rows of the same person in a certain company and keep the overall number of months the person worked there. That's because, in most cases, a person has a single row for every company (possibly with multiple positions).
10. Create columns to hold a person's history statistics such as average job duration etc.
11. Fill in missing values using Imputation. This process will be described in the next section.

3. Data Analysis

Our feature set contains a few subsets of features. The first subset includes the employee, the company, and the job title. The second includes experience history features, which contain the total, average, and last months a person worked. The next subset includes education features, indicators of having a bachelor's degree, a master's degree, or a doctorate. The last subset of features relates to the company review measurements of happiness, role rating, general company rating, meta-industry of the company, and size.

Handling missing values

The following columns had NaNs, and all the other ones were full:

Column	rating	happi-ness	roles	has_bachelor/master/doctor
%	0.2%	31.9 %	17.4 %	8.9%

We checked two approaches to handle the missing values correctly: assuming MCAR and removing any row containing NaNs or assuming MAR and imputing the missing values. To decide which approach is better, we had to perform the feature analysis with respect to each approach. The feature analysis is composed of calculating the correlation matrix between every pair of features, describing each feature by its minimum and maximum values, mean and standard deviation, pair-plot (a scatter plot of each feature with respect to every other feature), a boxplot for each category of each categorical feature describing the distribution of exp_months (target feature) over that category.

The results of the first approach are available in the appendix. We can see a noticeable correlation between happiness and the roles feature, and the target feature does not have any noticeable correlation with any other feature, which might hint at a non-linear dependency. People who work in the 'Government and Public Policy' meta-industry tend to stay in their company longer. The happiness measure is normally distributed around 68, whereas the roles' ratings follow a Pareto distribution in which most roles are considered 4.8 (out of 5).

For the second approach, we incorporated five imputation techniques: zero, mode, mean, constant, and linear regression. The first four imputation techniques didn't improve the correlation of any pair. Moreover, these four techniques impute the same value for all missing values. This kind of imputation usually reduces the standard deviation of the imputed features and, therefore, might introduce a bias. Following this logic, we understood these techniques were not fruitful, so we used regression imputation.

Regression Imputation

Since we wanted to impute all the missing values in all the features, we had to impute feature by feature iteratively. So, we start by imputing 'rating' since it has the least missing values while assuming a linear model between 'rating' and 'meta_industry' and 'employees' (both are categorical and therefore encoded as one-hot). Having done that, we've moved on to impute 'roles' while assuming a linear model between it and 'meta_industry', 'employees', and 'rating'. Only then did we impute the last feature, 'happiness', using all four features mentioned in this paragraph. As you can see here, these imputations had little to no effect on the general statistics of the features (compared to the statistics before the imputation).
This imputation technique allows us almost to double the data we had (from 50k to 100k rows) while minimizing the bias introduced by the imputation. You can view the full feature analysis for this imputation technique here.

Feature Selection

For feature selection, we decided to treat various feature sets as a hyperparameter. This implies that our models were trained and evaluated on different feature sets. After checking different subsets, we concluded that it is worth checking a linear combination of certain features. For a qualitative selection of the linear combination, we incorporated PCA on each feature subset (each of which is available in the results table). More specifically, for each feature set S, we took

the 0.5 * |S| most significant features to maximize the explained variance of the data.

Additional features

After observing the unsatisfying results of our model, we tried creating new features that will hopefully better handle the numerous outliers (extremely short or long job durations). We extracted the job titles and companies with the highest correlation with short/long job durations while having a low correlation with the other group of job durations (long/short, respectively). We considered the proportion of long vs short job durations as the weight of each feature.

4. ML & DL Methodologies

Our target column (Y) is "exp_months" - the number of months a person worked for a particular company. Therefore, we had to deal with a regression task. We examined a few baselines and tried to predict "exp_months" with various Machine Learning models. We split the dataset into a train set and validation set (80% and 20%, respectively) and made sure that the split is also done across subjects, so if a person appears in the validation set, he won't appear in the train set. That way, we can ensure the model is generalized across different people.

4.1 Baselines

1. **Random:** sample from a normal distribution, where the train set estimates the mean and std.
2. **Mean/Median:** fill in the mean/median Y from the train set.
3. **Mean/Median by Meta industry:** fill in the mean/ median of Y grouped by the meta industry from the train set.

4.2 Models

1. Linear Regression - including regularization.

2. XG-Boost - the best result was achieved with n_estimators=850, max_depth=7, learning_rate=0.01, gamma=0.64, subsample=0.94, colsample_bytree=0.96 (achieved by an "Optuna" study).

3. Fully Connected Neural Network - The best result (visible here) was achieved using all the features (input dimension of 1577) and four hidden layers of dimensions: 1024, 512, 256, and 128. On every layer, we applied drop-out of 0.2, ReLU activation, and batch normalization

Hyperparameters tuning: we experimented with different hidden layer dimensions, different numbers of hidden layers (2-4 layers), different activation functions (tanh, ReLU, sigmoid), different drop-out rates (0.2-0.4).

Additional Models (After Poster Presentation)

4. PCA: For each feature set that consisted of |S| features, we took the 0.5 * |S| principal components, as we explained earlier.

5. Fusion

We denote the prediction from the XG-boost model as v_x, the FC model as v_fc, and the new prediction as v'. Then, we define v' = 0.5 * v_x + 0.5 * v_fc.

4.3 Features Sets

Additionally, we examined a few approaches to prediction: (1) using only features of experience history (based on profiles data frame), (2) using only features of the company reviews (based on company review data frame), (3) using both features of experience history and company reviews without textual features and (4) using all features.

5. Evaluation and Results

5.1 Evaluation

Our evaluation metric is RMSE - the root mean squared error.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

$\hat{y}_i$ is the predicted number of months for employee i of his current job. y_i is the true number of months.

5.2 Results

At the first iteration of working on the models, the best result was 42.67 with the fully connected model and 42.97 with the XG-Boost model, both using all features. After discussing our ideas while presenting our poster and after further error analysis, we added more features (as mentioned before in the "Additional features" section). The new result of XG-Boost, including the new features, is 42.65, and 42.9 for the FC model. We tried additional models: the PCA and the Fusion. Their results were 43.1 and 42.8, respectively. Finally, the best result we got is 42.65 from the XG-Boost model. (all detailed results are presented in Table 2)

6. Limitations and Reflection

Predicting employment duration is a complex task due to the many factors that affect one's decision regarding job selection. Factors such as financial status, family status, socio-economic status, and career path desires are only some of the crucial aspects that affect one's employment duration. The data we used in this project can not account for these aspects. Therefore, it is fair to assume that with some additional features that can shed some light on these latent variables, we might be able to develop a model with a higher accuracy. Some of these latent variables could be partially explained by features such as salaries, ethnicity, employment type (full/part-time, freelancer, entrepreneur, contract worker, etc.), demographics, medical history, gender, and many more. If we look at the neural network loss graph on the validation set, we can see that the network did not learn the underlying distribution. This implies that either the model lacks expressive power or the features are not descriptive enough, which supports the notion that the features we collected are insufficient.

7. Conclusions

The best result we arrived at was an RMSE of 42.6, which means that, on average, our model misses one's company transition by 3.5 years. This result cannot provide significant insights to HR and job recruiters in the modern, fast-paced labor market since this period is far too long. Companies cannot rely on it to plan personalized retaining programs. They probably will not rely on the model's ability to correctly predict other companies' employees' transition to snatch the professionals to themselves. A significant takeaway from this project is that adding the BERT embeddings of the companies' names and the employees' roles noticeably reduced the RMSE. This might suggest that certain companies are more successful at keeping the talents satisfied at their workplace. Therefore, other companies can learn from them.

Appendix

Table 1. An example of the full table with the various features

PROFILES											COMPANY REVIEWS					
name	clean company name	exp title	exp months	total exp	avg exp	last exp	has bachelor	has master	has doctor	is intern	rating	happi-ness	roles	company count	meta_industry	employees
AVINASH T	anthem	Database Engineer	12	60	30	53	0	0	0	0	3.6	66	5.0	2	Healthcare and Medical	10,000+
AJ Aleccia	flagstar bank	Marketing Analytics Lead	94	72	72	72	1	0	0	0	3.2	57	4.3	1	Financial and Investment	1,001 to 5,000
Abby Hoffman	kent state university	NSF REU Research Intern	3	4	4	4	1	1	0	1	4.1	76	4.9	1	Education and Training	5,001 to 10,000
Adam Rieth, PhD	rti international	Chemist	20	10	10	10	1	1	1	0	3.5	64	4.5	1	Healthcare and Medical	1,001 to 5,000

Table 2. Results

Project Iteraion (1-before poster, 2-after poster)		1								2	
	Feature Set / Model	Baseline 1 (Normal)	Baseline 2.1 (Mean)	Baseline 2.2 (Median)	Baseline 3.1 (Mean)	Baseline 3.2 (Median)	Linear Regression	XG-Boost	Fully Connected	Fusion	PCA
1	"exp_months"	60.05	46.48	49.02							
	"exp_months" & "meta_industry"				46.05	48.46					
	Only experience history						45.75	44.98	45.25		45.74
	Only company review						45.24	45.48	45.31		45.89
	All non textual features						55.12	44.27	44.55		44.84
	All features						45.52	42.97	42.67		
2	All features + additional features							42.65	42.9	42.8	43.1

Below are additional notes and plots from the data analysis, the training and evaluating process, and the error analysis part.

Appendix **6**

Table 1. An example of the full table with the various features 6

Table 2. Results 6

1. Dataset 6
2. The full features list 7

Additional Features 7

3. General statistics of the features before and after regression imputation 8
4. Pairwise feature scatter plot 8
5. Distribution of exp_months 9
6. Distribution of the features 11
7. Train MSE Loss of FC Model by Epoch 13
8. Test MSE Loss of FC Model by Epoch 13
9. Train MSE Loss of FC Model by Epoch - with additional features 14
10. Test MSE Loss of FC Model by Epoch - with additional features 14
11. Error Analysis Plots 15

1. Dataset

Link to the dataset – [Company Reviews (kaggle.com)]

2. The full features list

- 'Name' (string): The employee's name.
- 'Clean company name' (string): The name of the company where the employee works.
- 'Exp title' (string): The job title of the employee in that company.
- 'Exp_months' (int): The target feature describes how long the person will work or has worked in that company.
- 'Total_exp' (int): The cumulative months the person has worked.
- 'Avg_exp' (float): A cumulative average of months the person has worked in each company.
- 'Last_exp' (int): The number of months the person has worked in the last company
- 'has_bachelor', 'has_master', and 'has_doctor' (binary): Specify the employee's degrees. An employee can't have a higher degree without having a more basic one. These fields might also include NaNs.
- 'Is intern' (binary) - Extracted from the employee's title.
- 'Rating' (float): The average between 'rating' (the rating users gave to a particular company) and 'ratings' (an average of various parameters such as work/life balance, culture, and so on), both of which range from 1 to 5.
- 'Happiness' (float): An average of various sub-features such as "Work happiness score", "Energy", "Trust", "Belonging", and so on. It describes the happiness of
- working at a specific company.
- 'Roles' (float): Each role in a company has a rating ranging from 1 to 5. Our 'roles' features are an average of all of these roles.
- 'Meta_industry' (categorical) is the meta-industry of the person's company, as described in previous homework assignments.
- 'Employees' (categorical) specifies the range of the company's employees (1, 2 to 10, 11 to 50, and so on).

Additional Features

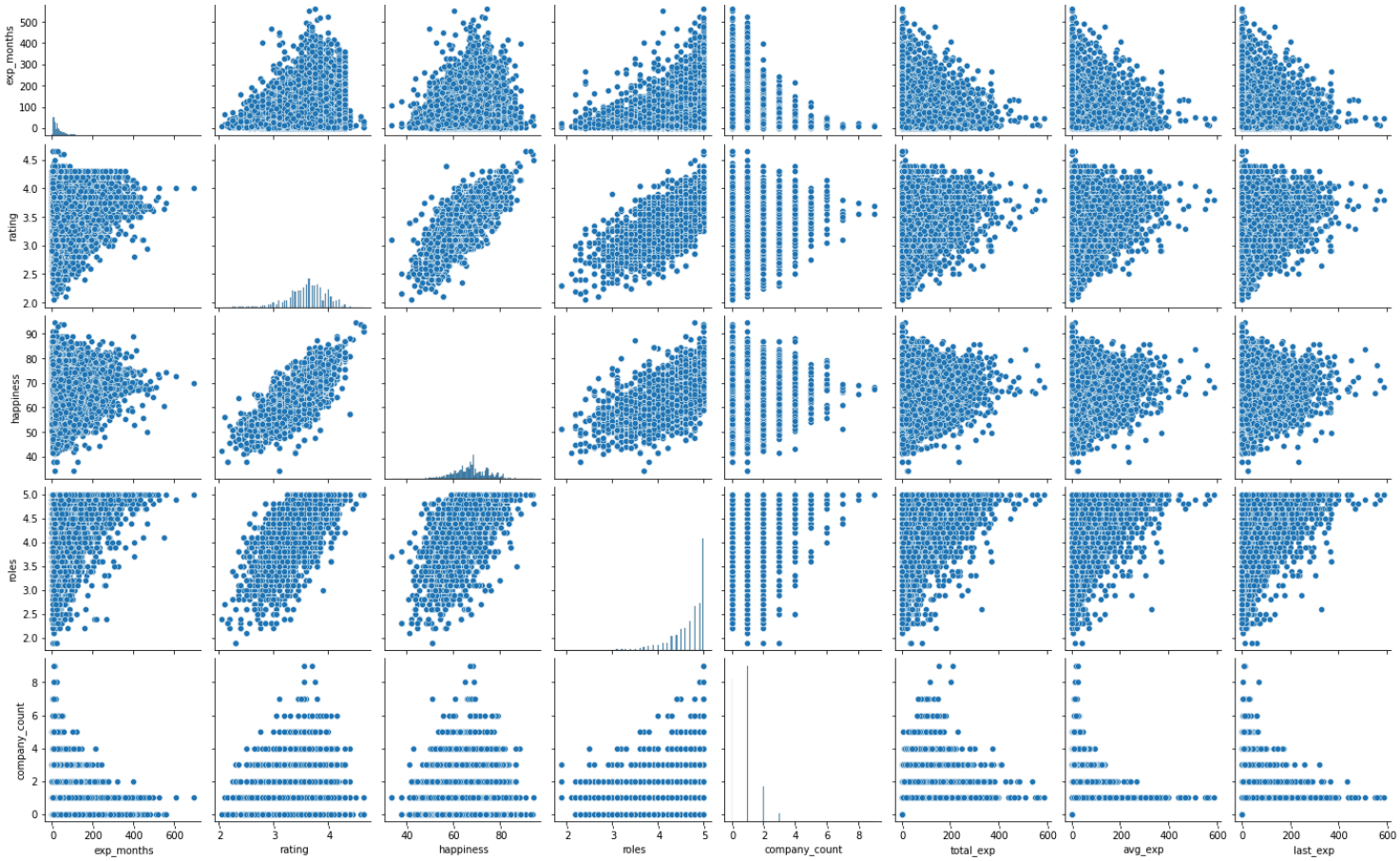
- 'over_title', 'over_comp', and 'under_title', 'under_comp' (float): The proportion of long vs. short job duration amount in a certain title or company.

3. General statistics of the features before and after regression imputation

Impution Col	happiness		roles	
Stats	before	after	before	after
% Null	**32%**	**0%**	**17%**	**0%**
mean	**67.80**	**67.80**	**4.65**	**4.63**
std	**7.01**	**6.99**	**0.42**	**0.41**
min	**27.50**	**21.45**	**1.80**	**1.80**
max	**94.40**	**95.60**	**5.00**	**5.00**

4. Pairwise feature scatter plot

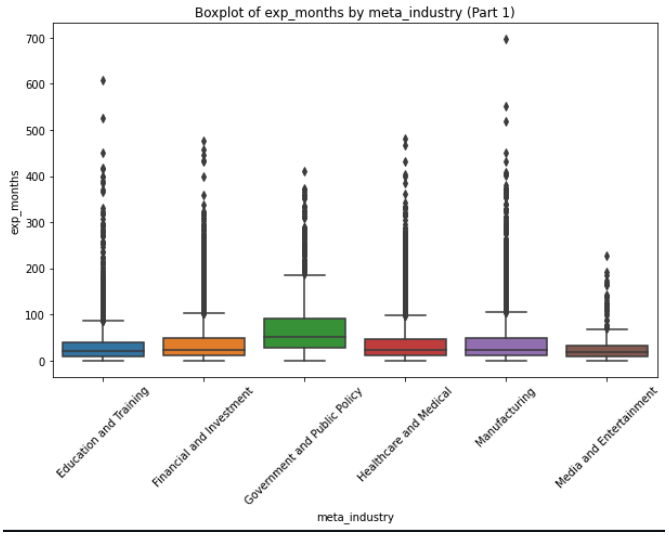
The following plots describe the data after the regression imputation was incorporated.

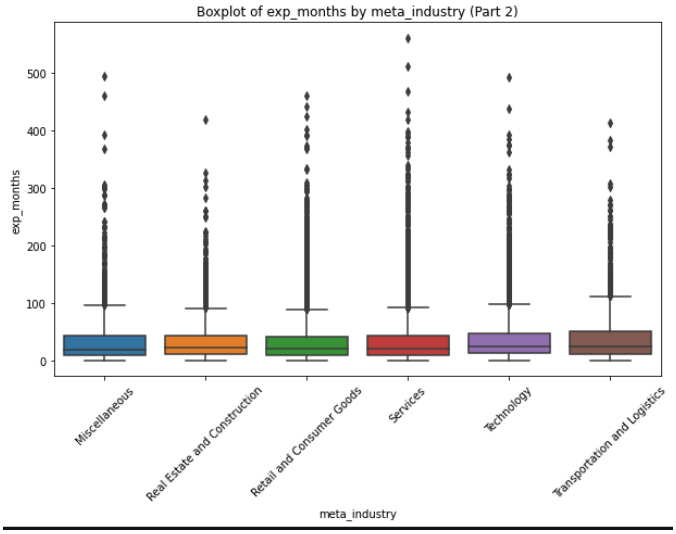


5. Distribution of exp_months

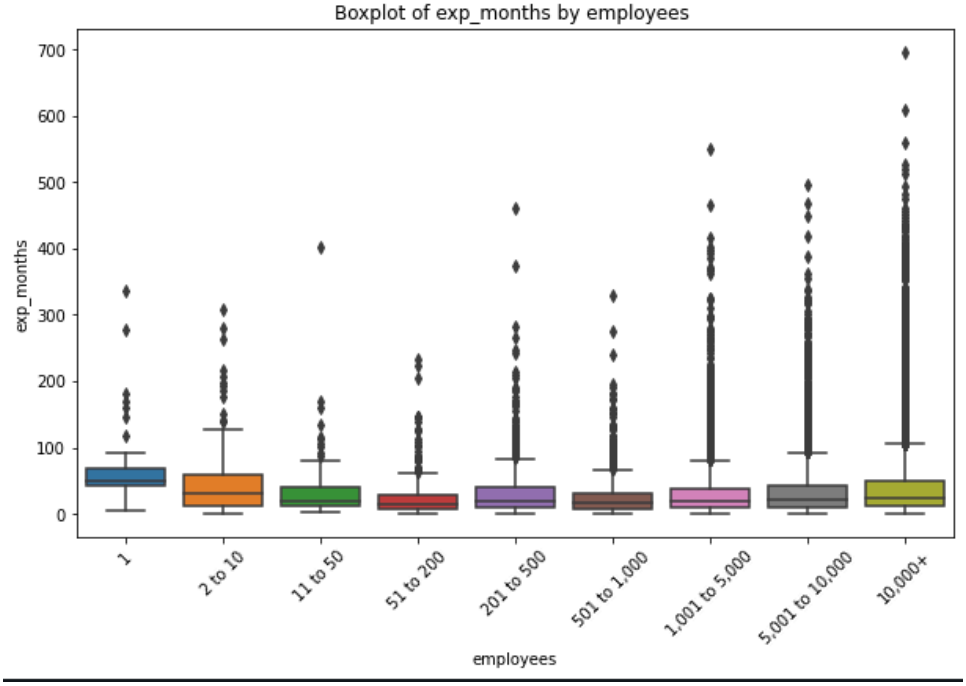
Plots across each category for each categorical feature

a. Exp months boxplot by meta industry

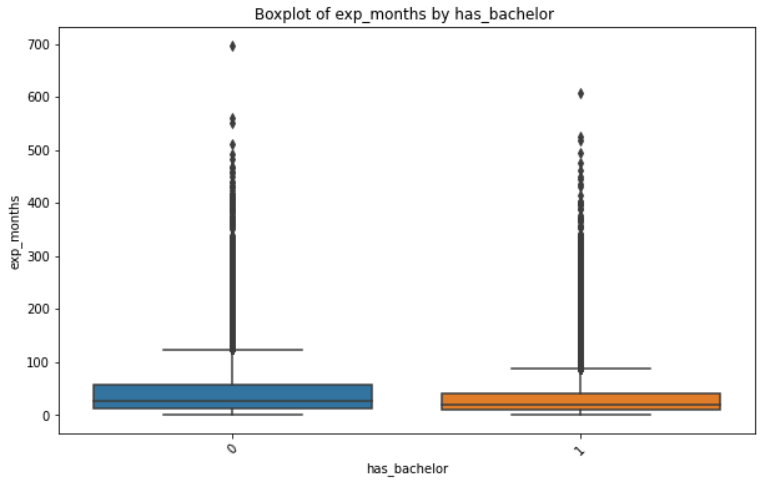




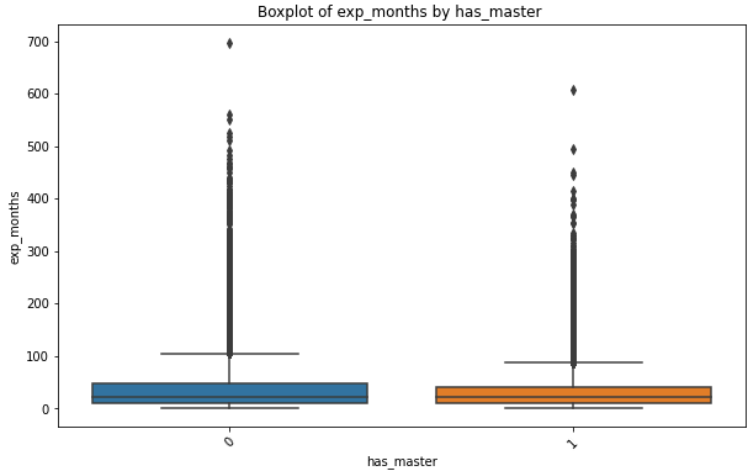
b. Exp months boxplot by employees



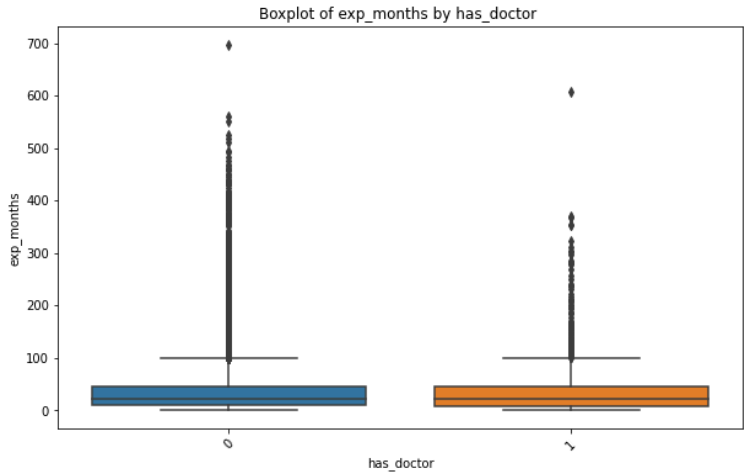
c. Exp months boxplot by has_bachelor



d. Exp months boxplot by has_master

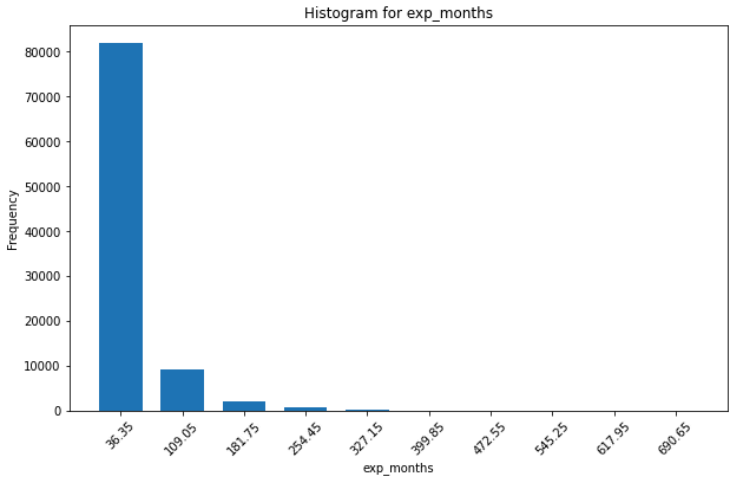


e. Exp months boxplot by has_doctor

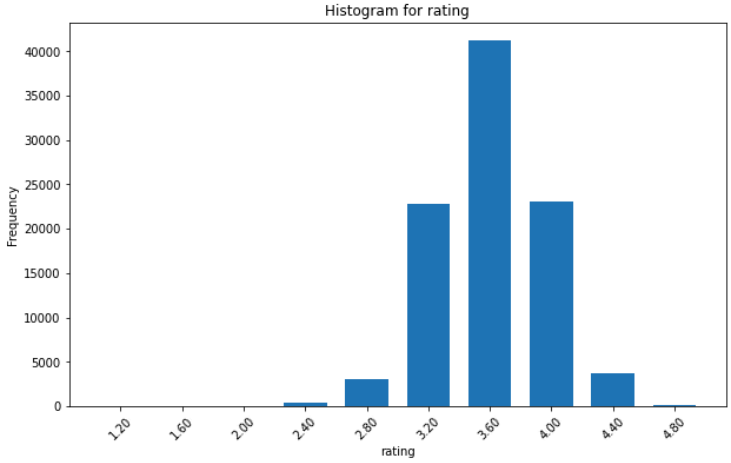


6. Distribution of the features

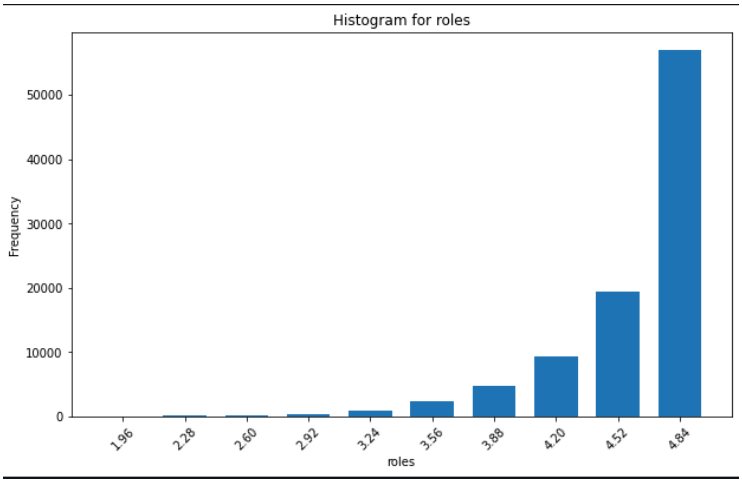
a. Histogram of exp_months



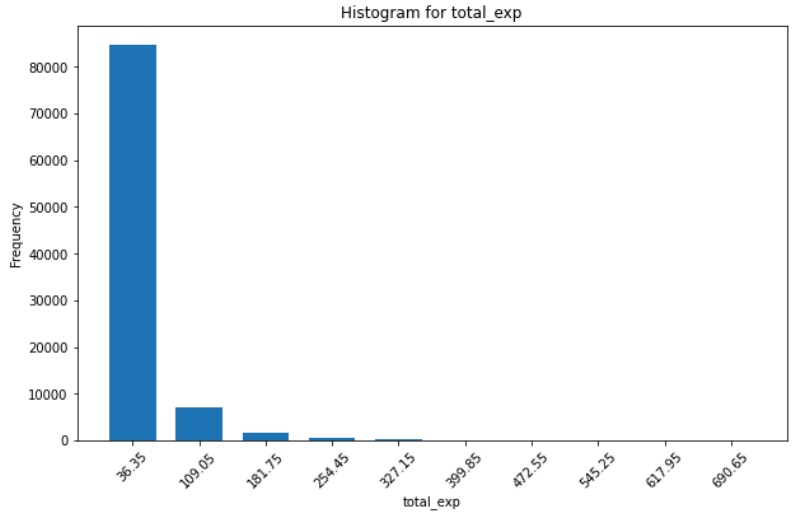
b. Histogram of rating



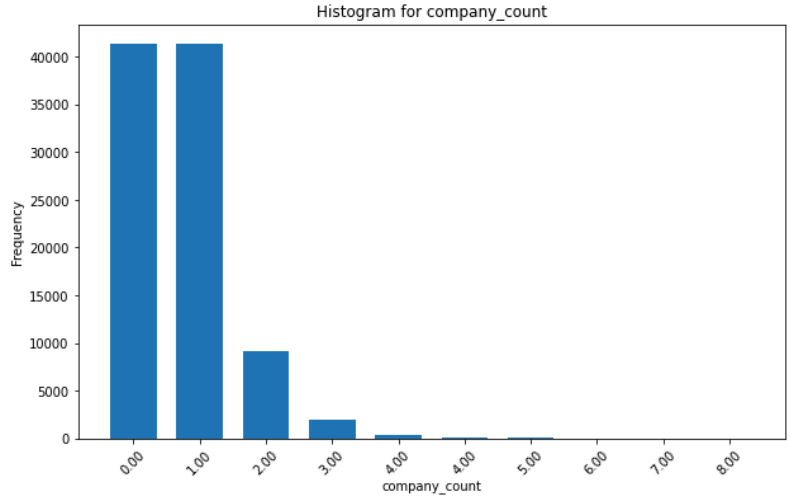
c. Histogram of roles



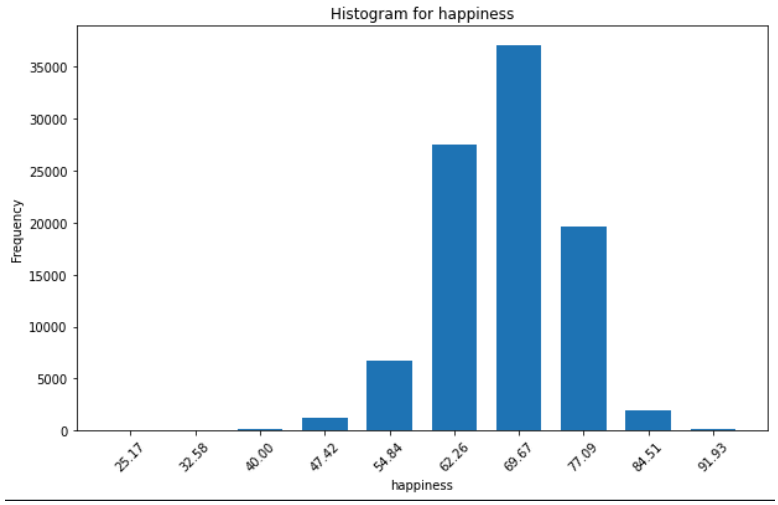
d. Histogram of total_exp



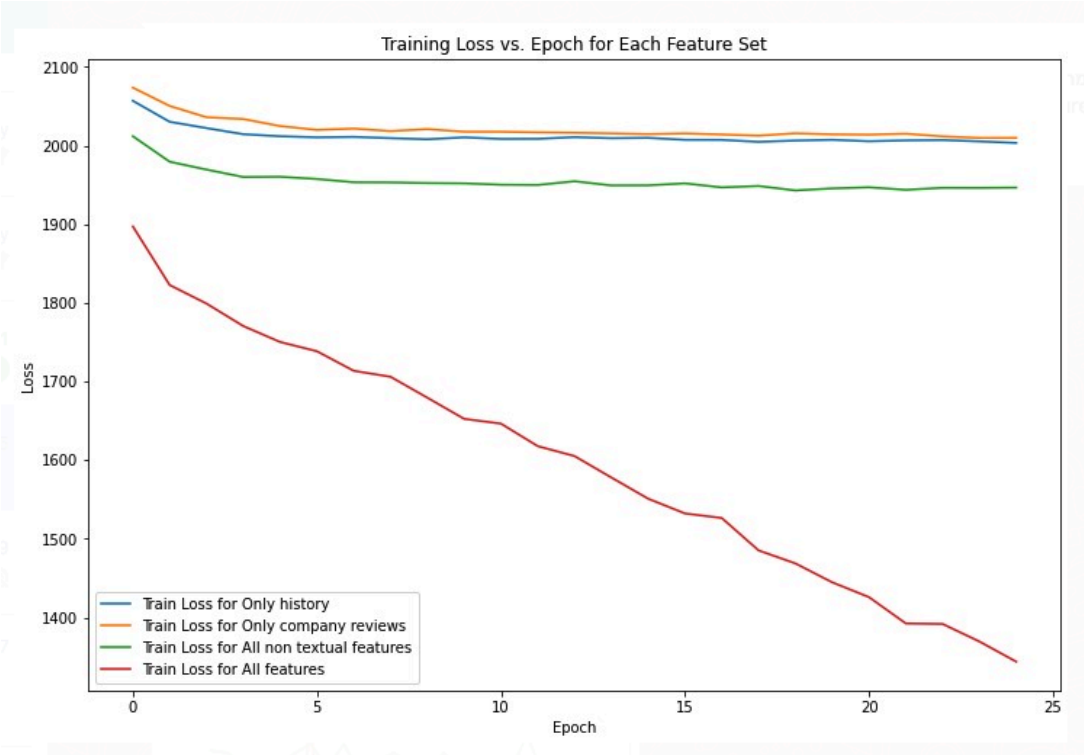
e. Histogram of company_count



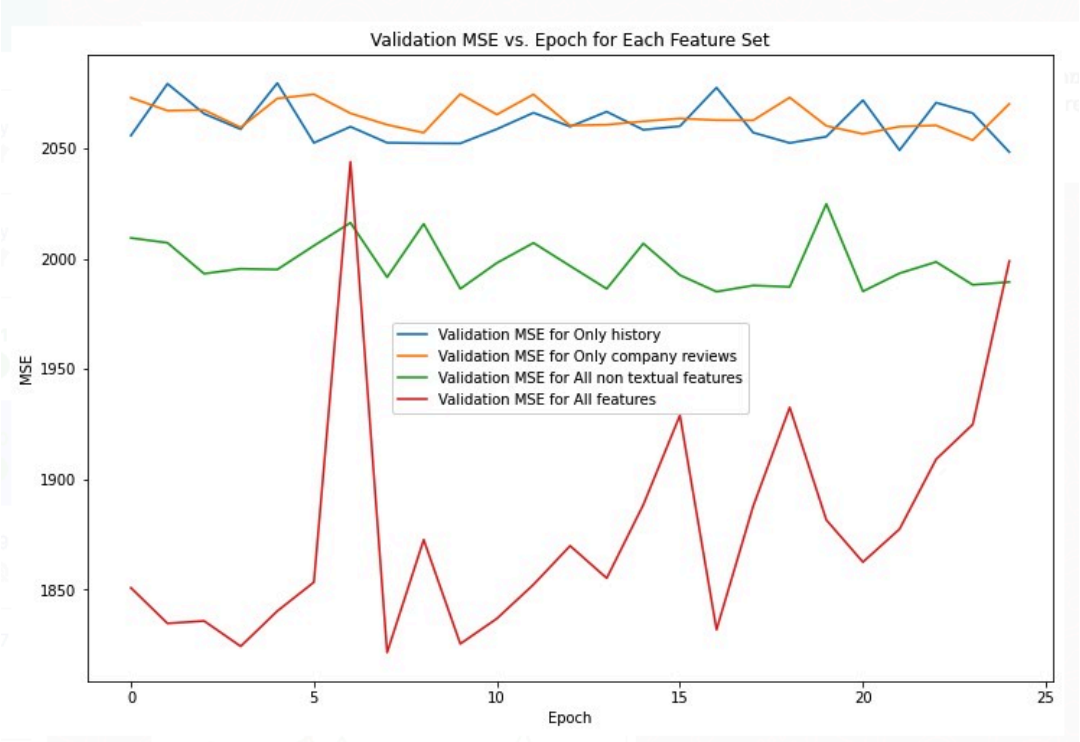
f. Histogram of happiness



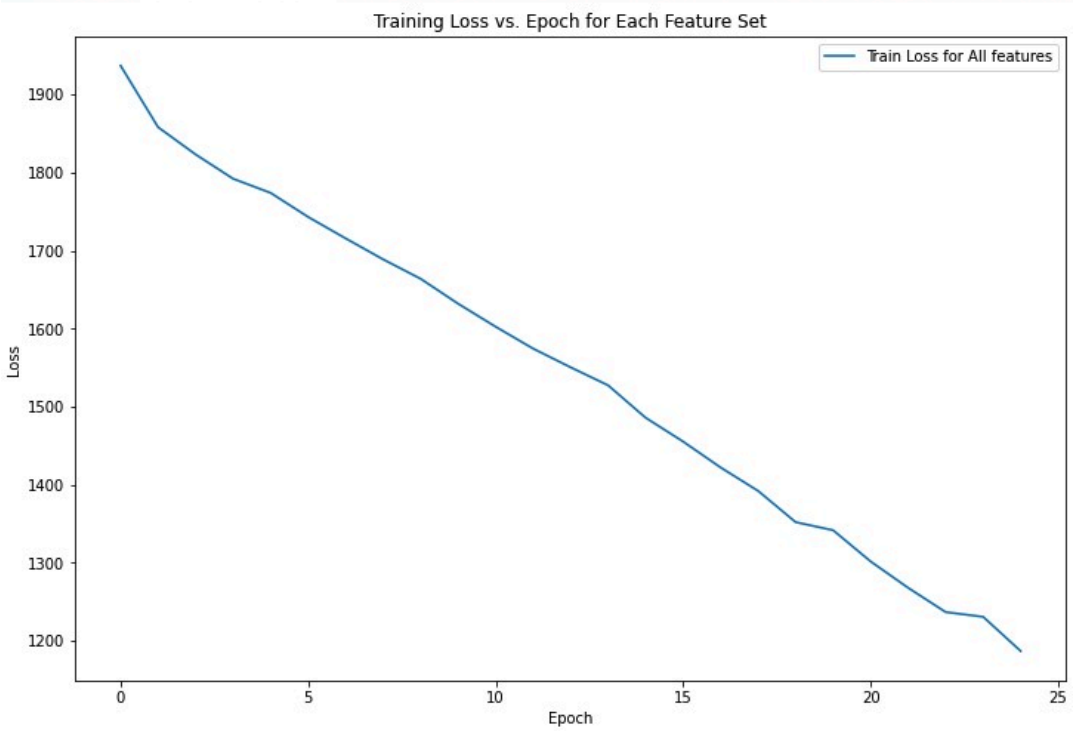
7. Train MSE Loss of FC Model by Epoch



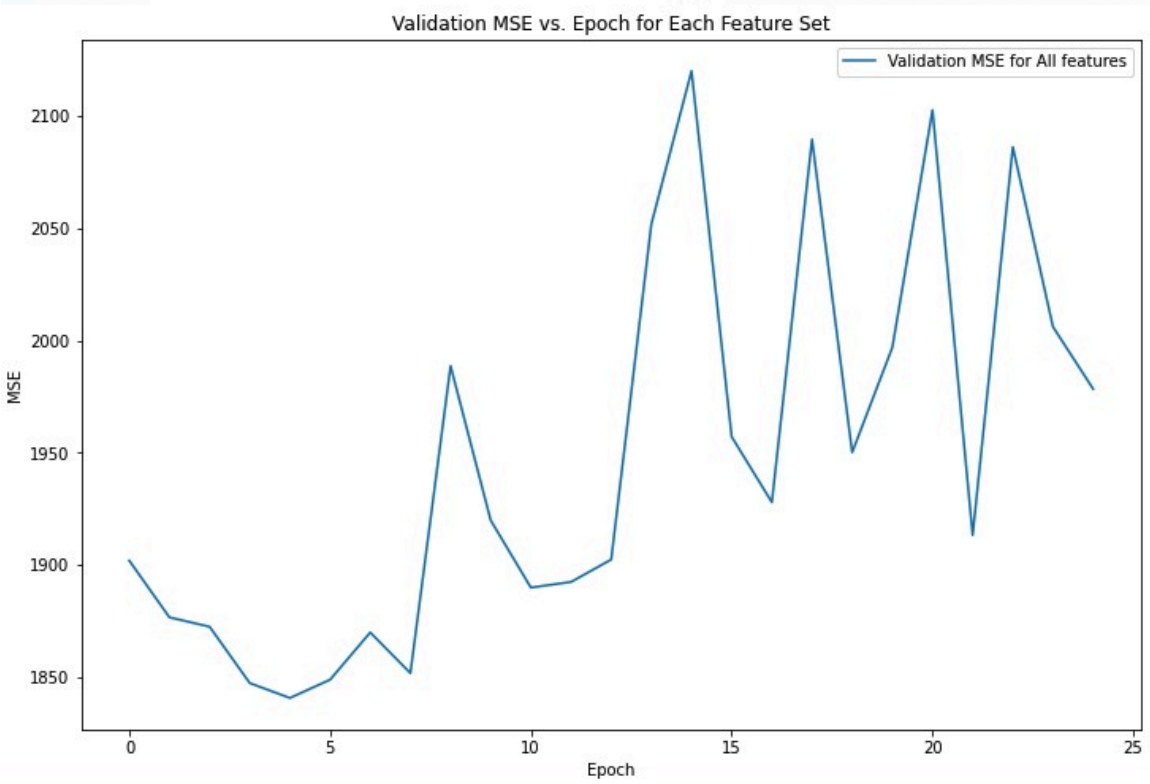
8. Test MSE Loss of FC Model by Epoch



9. Train MSE Loss of FC Model by Epoch - with additional features

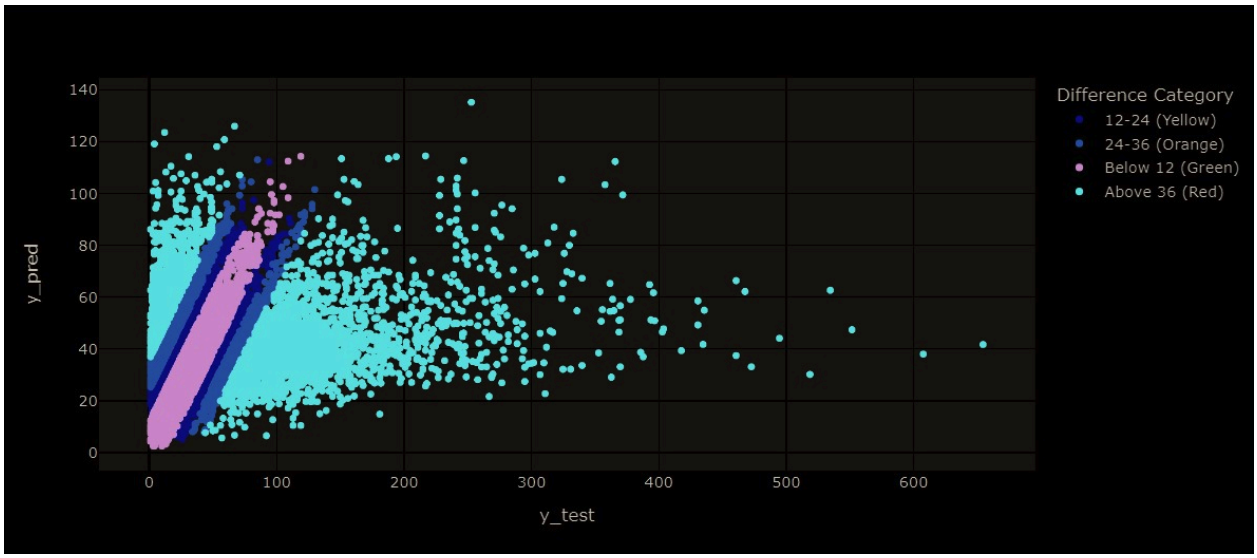


10. Test MSE Loss of FC Model by Epoch - with additional features



11. Error Analysis Plots

Below, we can see y_pred by y_test for the predictions from the XG-boost model. The different colors represent different categories of errors. Using this plot, we can easily see that the model predicts values closer to the mean Y of the train set as the y_test gets larger. Then, we started investigating features of employees with very high exp_months values (large Y values).



We then created the additional features - over/under_title/company features. Here, we can see examples of these features. For each company, we take the long vs short job duration proportion amounts (in terms of people). For example, in the US Army, the number of people who worked more than 100 months is ~5 times higher than those who worked less than 12 months.

	over_comp	propor
2	us army	4.976744186046512
3	jpmorgan chase	1.5769230769230769
4	department of veterans affairs	2.0833333333333335
5	us air force	6
6	the hartford	1.6
7	us coast guard	3.75
8	social security administration	1.625

16 rows | 8.36 seconds runtime

Table +

	under_comp	propor
1	walmart	3.784313725490196
2	wells fargo	2.569620253164557
3	bank of america	1.6228070175438596
4	microsoft	3.772727272727273
5	deloitte	9.11111111111111
6	google	3.864864864864865
7	best buy	5.2592592592592595

530 rows | 8.36 seconds runtime