# Attrition Analysis

Benjamin Edwards (bje43), Artina Maloki (am2495), and Lillyan Pan (ldp54)

**Abstract**

An analysis that identifies the strongest correlating features that relate to employee attrition and a model to identify potential warning signs at an employee level. For companies this will reduce incurred expenses to find replacements, recruiting, advertising, hiring and training, etc. and increase morale to boost employee productivity and the company's public image.

## 1. Exploratory Data Analysis

### 1.1 Data Characteristics

The *IBM HR Analytics Employee Attrition and Performance* dataset provides comprehensive information about employees, including demographic and financial data. The data set contains employee attrition data in the form of a variable matrix with 36 features and 1470 rows of employees. Each employee has a field specifying whether they left the company and how long they worked there. It also quantifies job satisfaction, work life balance and other company related metrics on a scale from 1 to 4, with 1 being low and 4 being very high. This dataset gives details into many possible causes for an employee to become unsatisfied with their current work situation. It also provides us with the granularity to see how factors contributing to employee attrition differ by age, gender and education.

The dataset is comprehensive and complete. We performed a data validation test to assure that there are no null or missing values. The features in the data set are broken down into the following data types,

- 17 continuous variables, including age, monthly income, and years at company.
- 8 categorial variables, including education level, job level, and department.
- 7 ordinal variables, including job satisfaction and work life balance.
- 4 binary variables, including employee attrition.

*Note, as this dataset was generated by IBM (due to privacy reasons) this dataset has been explicitly okayed by Prof. Udell.*

### 1.2 Data Visualizations

To get an overview of how the features are related to one another we created a correlation matrix on the continuous and ordinal variables. After visualizing results on a heatmap (Figure 1) we can observe that the majority of the data is uncorrelated, an advantageous attribute in creating our predictive model.

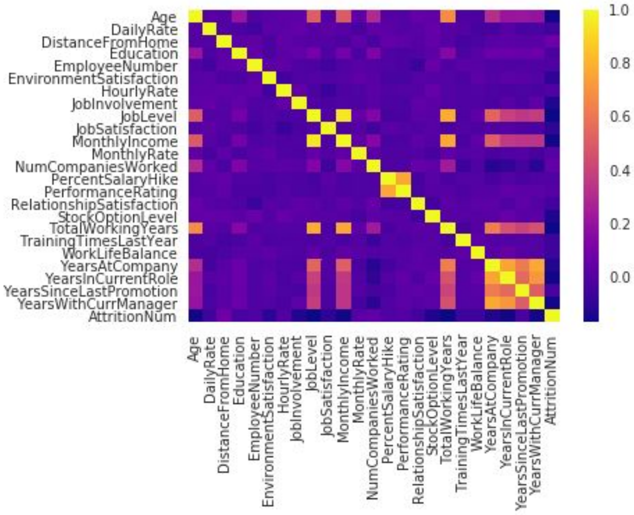

**Figure 1:** Correlation Matrix between Continuous and Ordinal Features

To analyze the distributions of the features in the dataset and their relationship to attrition we created several histogram and density plots. We plotted the distribution for three continuous features that we thought were interesting using density plots: *Age, MonthlyIncome*, and *DistanceFromHome* (Figure 2). From these plots we realized that these selected features will be important for our classifications since the distributions between the employees who left and those who stayed are dissimilar. For this specific company we discovered that on average the employees lost are younger, have lower monthly incomes than those who stay, and have longer commute distances.

We were also interested in the following categorial and ordinal features: *JobLevel, JobInvolvement*, and *DistanceFromHome.*

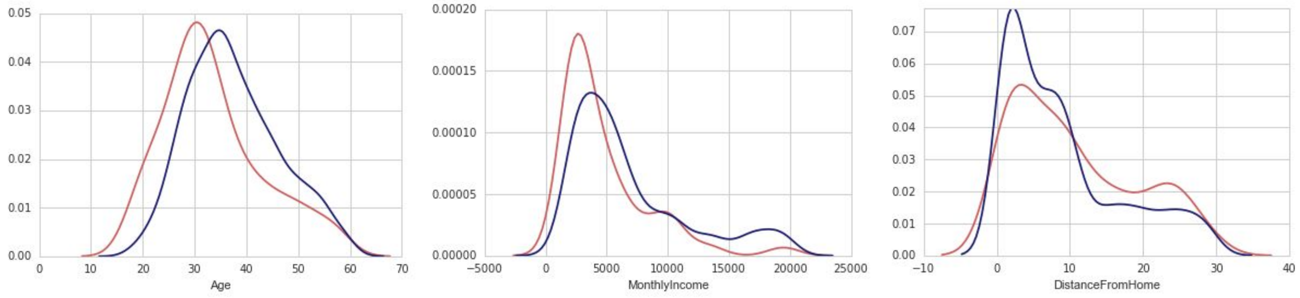

**Figure 2:** Distribution of Age, Monthly Income, and Distance from Home.

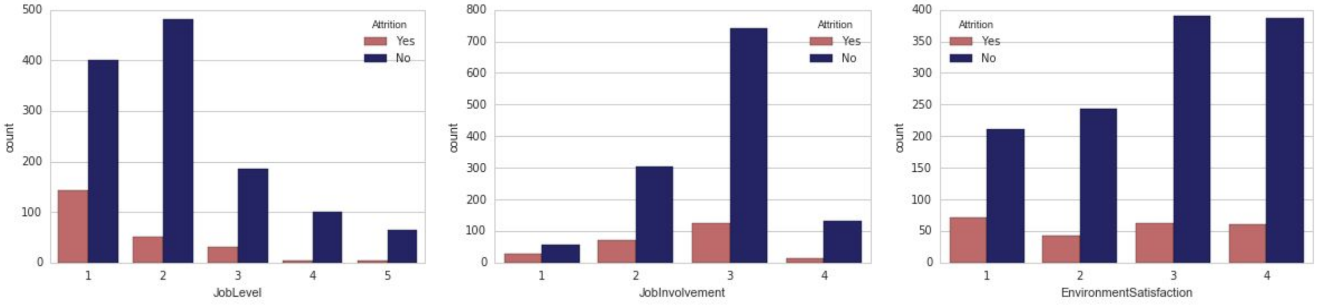

**Figure 3:** Distribution of Job Levels, Job Involvement and Environment Satisfaction

By plotting their distributions (Figure 3) we discovered that employees lost in attrition work in lower job levels, are less involved in their jobs and are less satisfied with their work environment than those who stayed.

### 1.3 Data Cleaning

The dataset contains several categorical and binary features, such as *JobRole* and *OverTime*. We used one-hot encoding to change the categorial variables into numeric values to allow for regression (expanding the number of features to 55). Similarly, we used indicator encoding to assign the binary variables to integer values. No data values were missing or corrupted as NULL or N/A values were checked for and summary statistics were run on the dataset to find any invalid data entries (ie. outliers).

## 2. Model Selection

### 2.1 Random Forest

A popular bagged algorithm is Random Forest, which is a collection of bagged decision trees with an altered splitting criteria. . The algorithm works as follows:

1. $m$ datasets are sampled from the original dataset $D$ with replacement
2. For each generated dataset $D_i$, a decision tree is trained to a max depth $p$ and before each split a subsample of $k \leq d$ features are only considered from this split.
3. Final classification is chosen using a majority vote among the trees.

Our Random Forest model used 50 trees for the forest and each tree was restricted to a depth of 30 (out of 55 features). Gini impurity was selected as the impurity function (as opposed to Entropy) as computationally intensive logarithmic functions are not required. A grid search will be used for optimal parameter search (in terms of tree depth and number of trees in the forest). Random Forest gives great flexibility to account for all characteristics of an employee with minimal preprocessing. Random Forest decreases variance by before each split a random subsample of features are only considered for the split– in our implementation we consider the square root of the total number of features at each split.

### 2.2 Logistic Regression

A regularized logistic regression model was implemented as another classification method. Logistic regression was used as opposed to other regression models because determining attrition in employees is a binary classification problem and other types of regression are meant to predict real valued outcome variables. A regularized version of logistic regression was used in order to reduce overfitting and to prevent the model from depending on every feature. The regularization term added to the normal logistic regression loss function was $||w||_2$ . A regularization parameter of $\lambda = 1$ and a L2-norm penalization were chosen both for simplicity and to reduce coefficient sparsity. In later analysis these parameters will be more accurately chosen.

The three features with the most impact on the model's prediction were MaritalStatus_Single, Age, and Daily_Rate (daily salary). MaritalStatus_Single was positively weighted while Age and Daily_Rate were negatively correlated. These are logically reasonable results as most people would expect older employees who are paid well to be less likely to leave a company than younger employees without family commitments. Interestingly, the BuisnessTravel features were given almost zero weight.

## 3. Results

Accuracy of our models were calculated based off the percentage in which Attrition was correctly classified. A summary of our findings are presented in Table 1. The fact that for the random forest model training accuracy was high when test accuracy was lower indicates that the model is suffering from high variance. This can be treated by adding more training data, reducing model complexity, and using bagging techniques. The training accuracy of the logistic regression model being somewhat low may suggest that the model is underfitting and the regularization parameter may need to be lowered.

| **Model** | **Training Accuracy** | **Test Accuracy** |
|---|---|---|
| **Random Forest** | 1.0 | 0.857 |
| **Logistic Regression** | 0.889 | 0.864 |

**Table 1:** Accuracy Results from Various Model Testings

## 4. Overfitting and Underfitting Treatment

In underfitting, the classifier learned on the training set is not expressive enough to account for the data provided as it does not account for relevant information present in the training set. This situation will exhibit both high training and test error. This is often resolved by trying a different algorithm that better applies to the data. In overfitting, the classifier learned on the training set is too specific and will not generalize to new data well. While training error will be low, test error will be high and increasing as the classifier cannot be used to infer anything about unseen data. To treat overfitting Early Stopping will be implemented– in which we will stop optimization after $M > 0$ gradient steps, even if the optimization has not converged yet. Note, this is already implemented in Random Forest by limiting the depth of the trees and that before each split a random subsample of features are considered for the split. Further, in Random Forest, overfitting is treated by taking a majority vote overall all classifications of trees in the forest. Cross validation will also be used to train and test the model on different subsets of training data and further estimate performance over different models on new data. Cross validation will also allow for an accurate choice of regularization parameter and p-norm to be chosen for the logistic regression model without overfitting to our test model. This will be done by re-partitioning the data to form include a validation set which will be used to determine the parameters for logistic regression after the models are trained.

## 5. Next Steps

Next steps include implementing Leave One Out Cross Validation (LOOCV) in order to better tune and compare our models. By using LOOCV scores, we will use a telescopic parameter search. Telescopic search will be performed using two searches: 1st, finding the best order of magnitude for $\lambda$; 2nd, performing a more fine-grained search around the best $\lambda$ found so far. Parameters for Random Forest include the number of trees in the forest and the maximum depth of the tree. To treat oversampling in the data, we will use SMOTE which creates synthetic samples from the minor class. Other next steps include fitting alternative models to the dataset such as SVM and implementing the improvements to the existing models listed under Section 3. Incorporating additional datasets and creating a model that will accurately predict the number of years an employee will stay at the company may be considered if the previous next steps are completed.