User IDs	12,000	
Users logged in	8823	74%
Users with 3+ logins	2248	19%
Adopted users	1597	13%

After finding which users became adopted users, I realized that the small number of positives would be something to keep in mind. Although the full email cannot be used, the domain may be useful. There were 6 domains that accounted for 90% of users and the rest of the domains only appeared once or twice so I grouped them as ‘other.’ Before training the classifiers, I removed the name, email, and date columns.

I then split the data and trained a logistic regression and a random forest classifier. The logistic regression classifier did not predict that any user would adopt and the random forest predicted a 6% adoption rate, but had mostly false positives. I thought this might be due to the relatively low number of adopted users so I used synthetic minority oversampling technique (SMOTE) to balance the training data. This caused the logistic regression to over-predict adoption and only slightly improved the random forest. I then removed the email domain column; the logistic regression classifier stayed the same and the random forest improved slightly.

Although one classifier has many false negatives and the other many false positives, the organization is the most important feature in both. Below is the code used for splitting the data and training the models.

```
X = df.iloc[:, :-1]
X = pd.get_dummies(X, drop_first=True)
y = df.iloc[:, -1]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=42)
sm = SMOTE(random_state=42)
X_train, y_train = sm.fit_resample(X_train, y_train)

lr = LogisticRegression(solver='lbfgs')
lr.fit(X_train, y_train)

rf = RandomForestClassifier(n_estimators=100)
rf.fit(X_train, y_train)
```