Data Science Practicum Report
Mar 20 th 2017

Preprocessing

As agreed in the meeting on Thursday 23rd, in addition to remove columns with >80% missing values, we kept columns suggested by Joel in the excel file sent on Monday Feb 27th. Our final list of 79 features was as follows:

	ECCNote	Eparent_welcome
EventsAttended	ECCNote_camp	Epost_event_email
Hprofileview	Ecoach_list_known_updated	Esms_update
Hcoachimport	ECoachEmailOpen	CollegeProspects
Hmessage	ECoachEval	MessagesReceived
Hsearchhit	ECoachImport	MessagesSent
Hcoacheval	ECoachSearchHit	CaptainU_CHURN
Hemailopen	ECoachVisit	NumYear
EAthlete newsletter	Ecolleges_going_to_the_event	NumMonth
Eathlete_new	Efailed_subscription	monthly_price
Eathlete_new_info_request	EEmailsDigest	Eparent_new
Gender	Hprofileview_Freq	Hcoachimport_Freq
Hmessage_Freq	Hsearchhit_Freq	Hcoacheval_Freq
Ecolleges_going_to_the_event_Freq	Efailed_subscription_Freq	Esms_update_Freq
Hits_Frequency	College_Prospects_Frequency	
Hemailopen_Freq	Ecoach_list_known_updated_Freq	ECoachVisit_Freq

We then took columns with continuous values and normalized the values by the mean and the standard deviation:

$$z = \frac{x - \mu}{\sigma}$$

Hence each feature had a mean of 0 and a standard deviation of 1. Below is a snapshot of the data:

```
In [379]: std_pd.head(5)
```

Out[379]:

Hprofileview	Hcoachimport	Hmessage	Hsearchhit	Hcoacheval	Hemailopen	EAthlete newsletter	Eathlete_new	Eathlete_new_info_request	...
-0.334910	-0.082896	-0.118345	-0.863017	-0.034355	-0.132922	-1.570523	-0.050501	-0.012864	...
-0.334910	-0.082896	-0.118345	-0.063949	-0.034355	-0.132922	1.506564	-0.050501	-0.012864	...
0.663747	0.843619	0.773367	1.001475	-0.034355	-0.132922	0.480869	-0.050501	-0.012864	...
-0.334910	-0.082896	2.556791	-0.197127	-0.034355	4.445944	0.480869	-0.050501	-0.012864	...
-0.334910	-0.082896	-0.118345	-0.330305	-0.034355	-0.132922	0.480869	-0.050501	-0.012864	...

We then created dummy variables from features with categorical features: gender .

We built models ignoring the sport a student plays.

Final table shape: **16117 rows 57 columns**

Predicting Churn in Girls

This week, we focused on predicting lifetime churn. We implemented three models machine-learning models recorded and visualized Precision and Recall Values.

Below are the three models we chose:

- Decision Trees
- Logistic Regression
- Support Vector Machines (SVM)

When it came to building models, we used 80% of the data for training and the remaining 20% for testing.

Training data: **7020 rows 57 columns**
Test data: **: 1755 rows 57 columns**

Class distribution of Churners and Non-Churners

Training set

Non-Churners	**4060**
Churners	**2960**

Testing Set

Non-Churners	**1006**
Churners	**749**

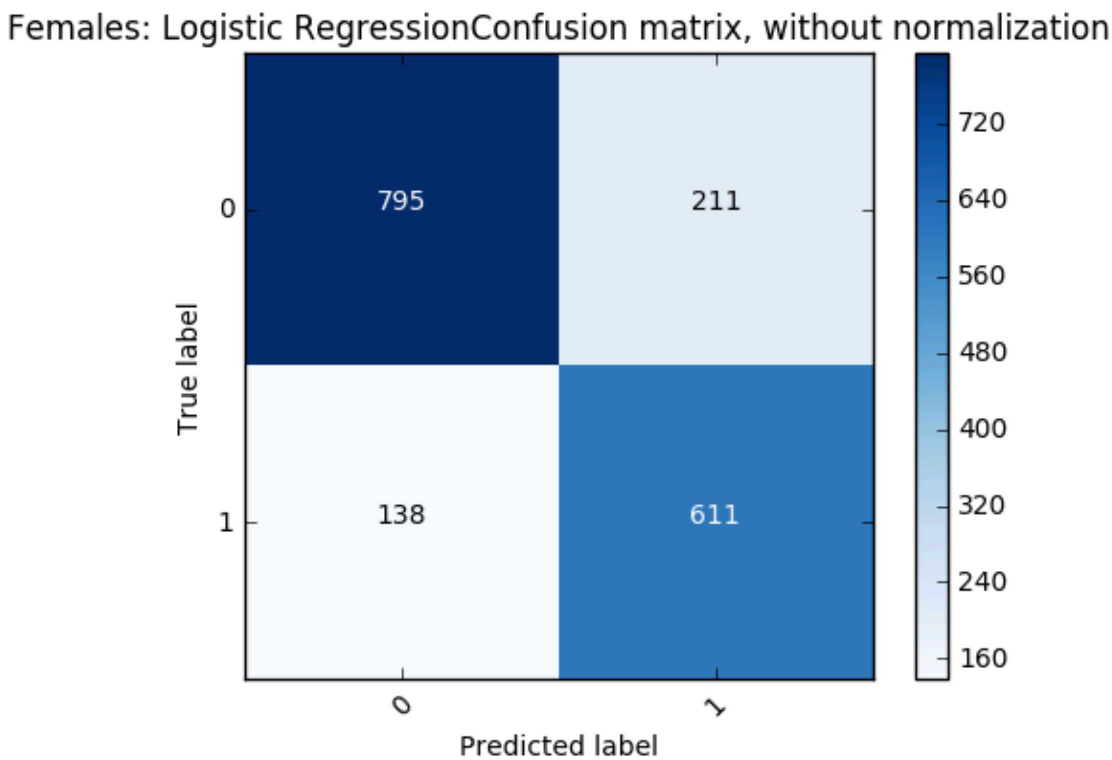
Summary of Precision Recall Values of Churning

Model	Precision	Recall	F1 Score
Decision Trees	0.71	0.73	0.72
Logistic Regression	0.74	0.81	0.78
SVM	0.73	0.83	0.78

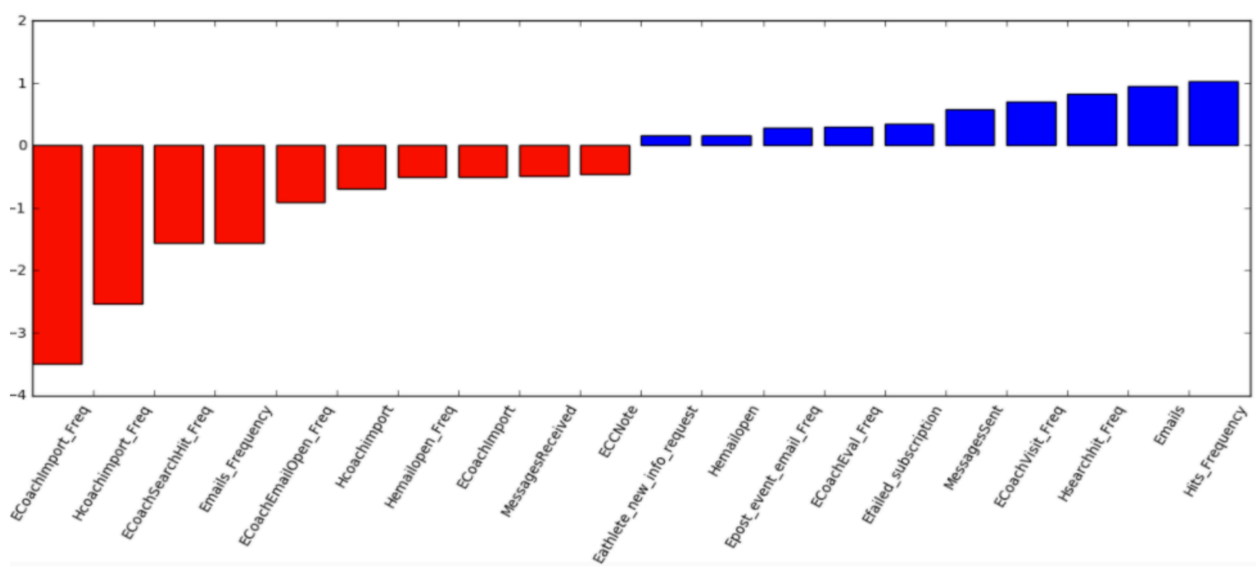
Summary of Precision Recall Values of Retention

Model	Precision	Recall	F1 Score
Decision Trees	0.79	0.78	0.78
Logistic Regression	0.85	0.79	0.82
SVM	0.86	0.79	0.82

Logistic Regression Visualization



Important Features According to Logistic Regression



Predicting Churn in Boys

Summary of Precision Recall Values of Churn

Model	Precision	Recall	F1 Score
Decision Trees	0.67	0.65	0.66
Logistic Regression	0.74	0.63	0.68
SVM	0.72	0.62	0.67

Summary of Precision Recall Values of Retention

Model	Precision	Recall	F1 Score
Decision Trees	0.84	0.83	0.83
Logistic Regression	0.83	0.89	0.85
SVM	0.83	0.89	0.85

Class distribution of Churners and Non-Churners

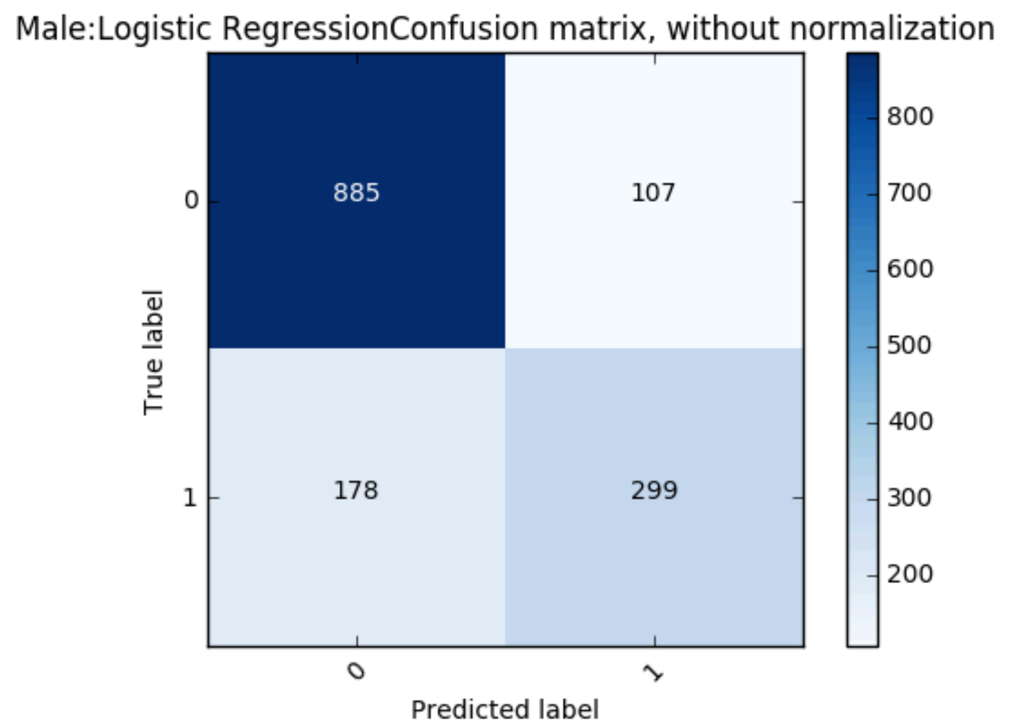
Training set

Non-Churners	**4016**
Churners	**1857**

Testing Set

Non-Churners	**992**
Churners	**477**

Logistic Regression Visualization



Important Features According to Logistic Regression