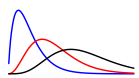
User activity prediction

Roman Lykhnenko

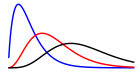
Introduction

- Goal: predict whether a user will buy something within 30 days from **now** for a user who made at least one order
- Data: customers of **Wine in Black**
- Used language: Python (Scikit-learn, Pandas, . . .)



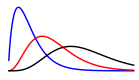
Basic concepts

- User is considered to be active if user purchased something within 30 days from **now**
- For the testing purposes: **now** is specified as 2015.06.01, and only users registered between 2015.01.01 and **now** have been considered.



Such features have been used for classsification of users

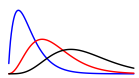
- Number of orders till today
- Frequency = (today - registration)/ orders
- Recency = time since last order till today
- Average order value
- Lifetime = days since registration (till today)
- Days between registration and confirmation
- Days between registration and first order

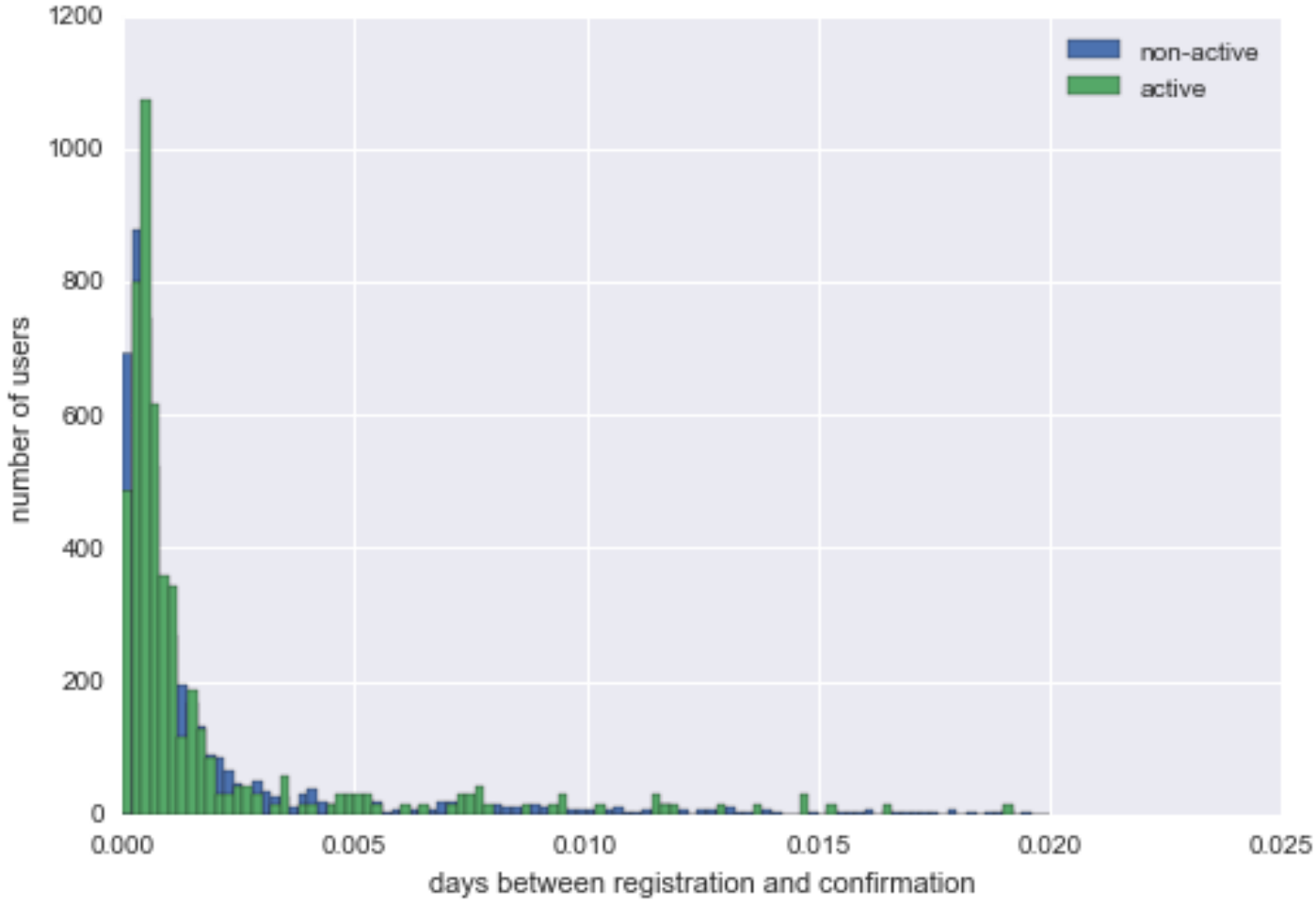


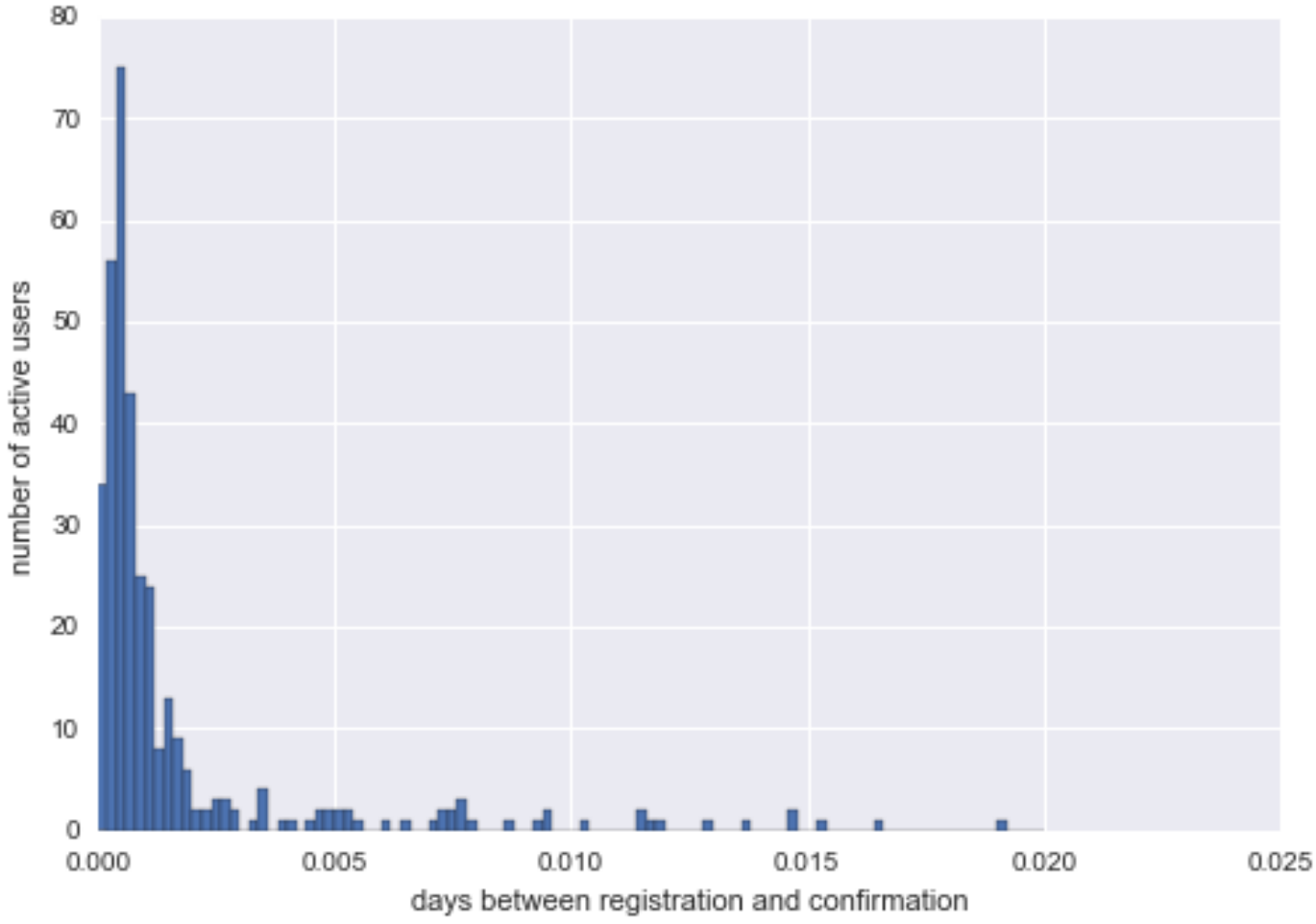
Graphics reveal data

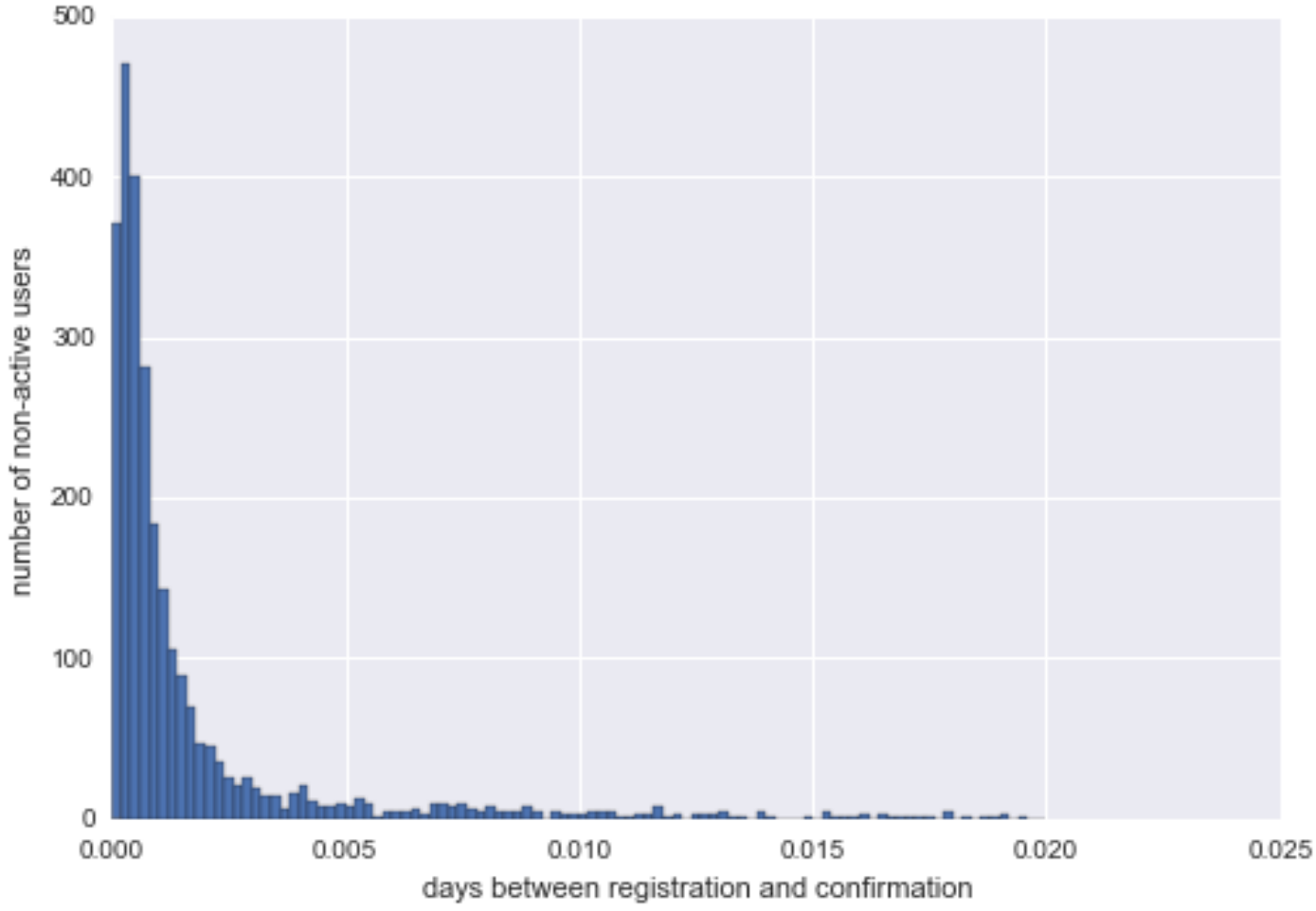
Plots below are supposed to shed light on the relation between features and activity of a user

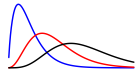
It would be great to have a clear separation of active and non-active users in the n-dimensional space of features.



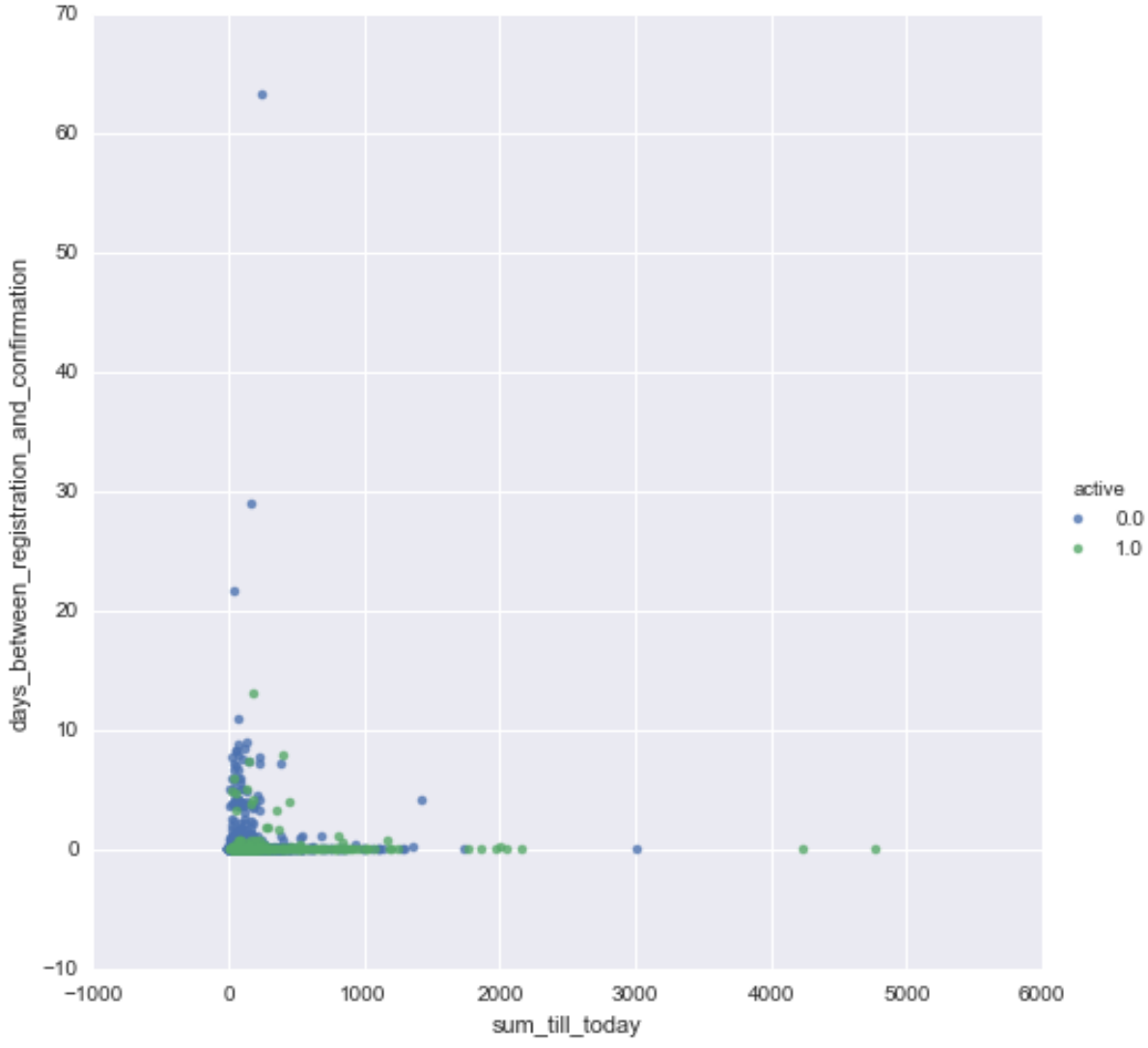

non-active
active
number of users
days between registration and confirmation
0
200
400
600
800
1000
1200
0.000
0.005
0.010
0.015
0.020
0.025

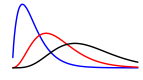

80
70
60
50
40
30
20
10
0
number of active users
0.000
0.005
0.010
0.015
0.020
0.025
days between registration and confirmation

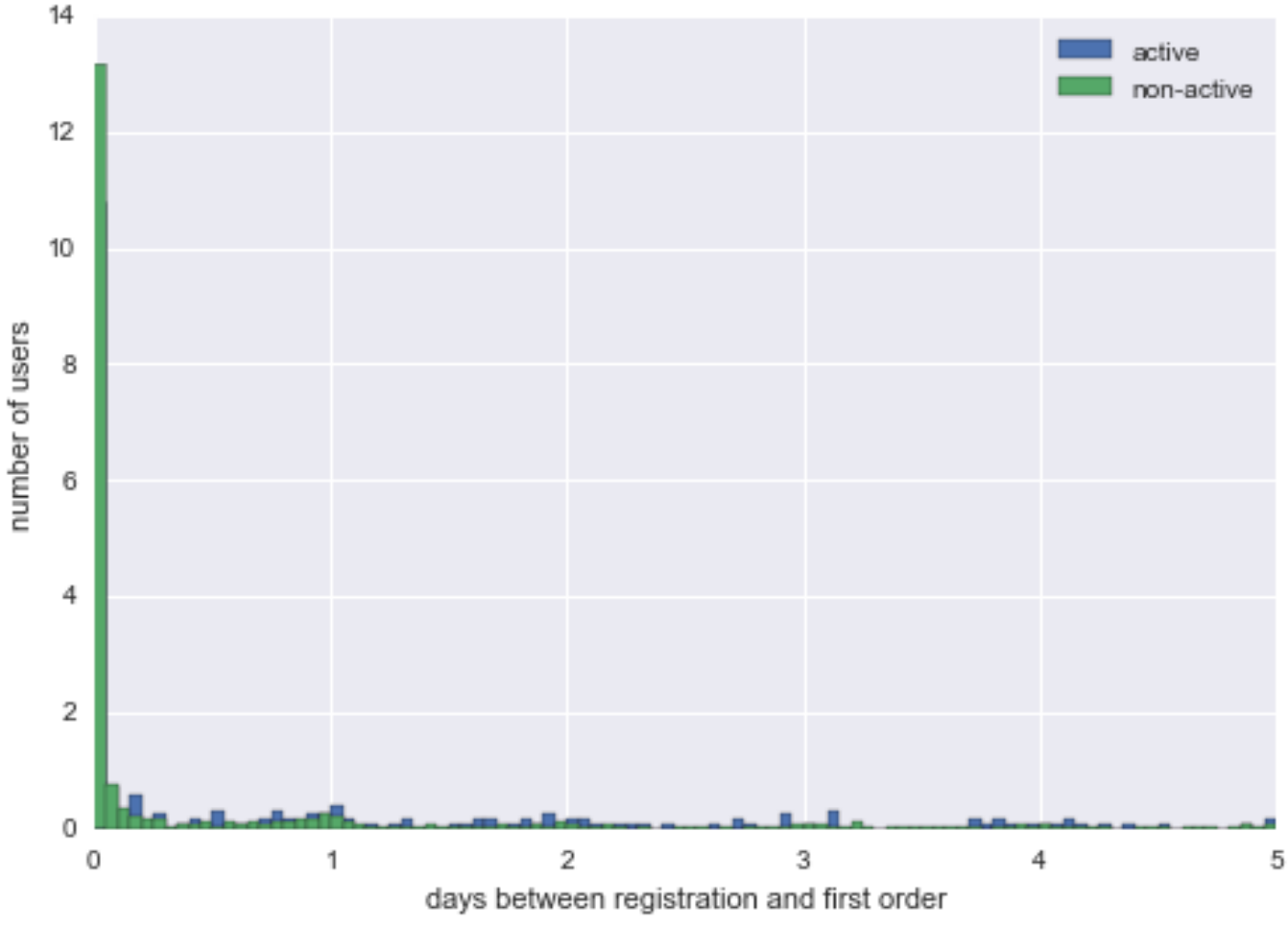

500
400
300
200
100
0
number of non-active users
0.000
0.005
0.010
0.015
0.020
0.025
days between registration and confirmation

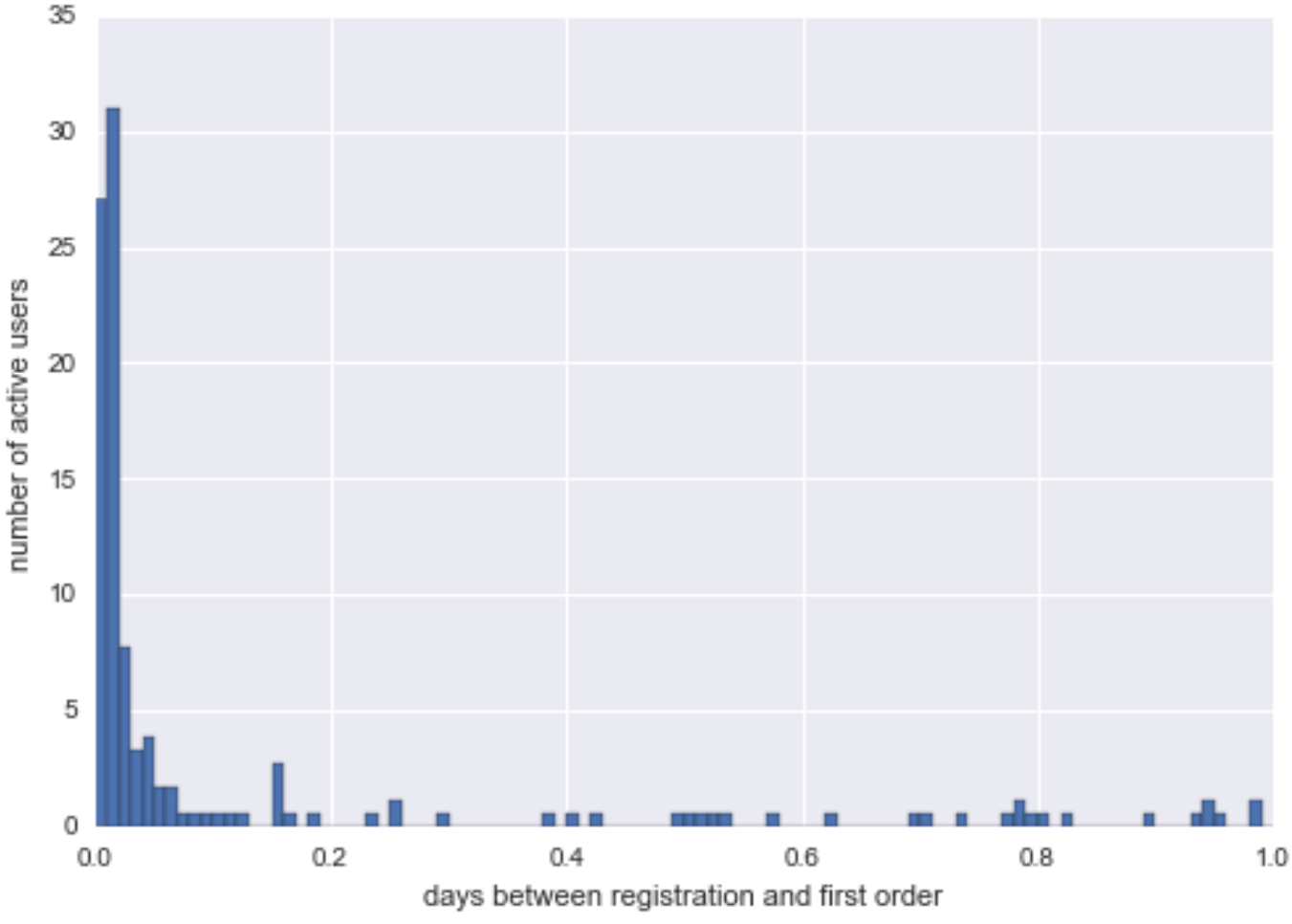


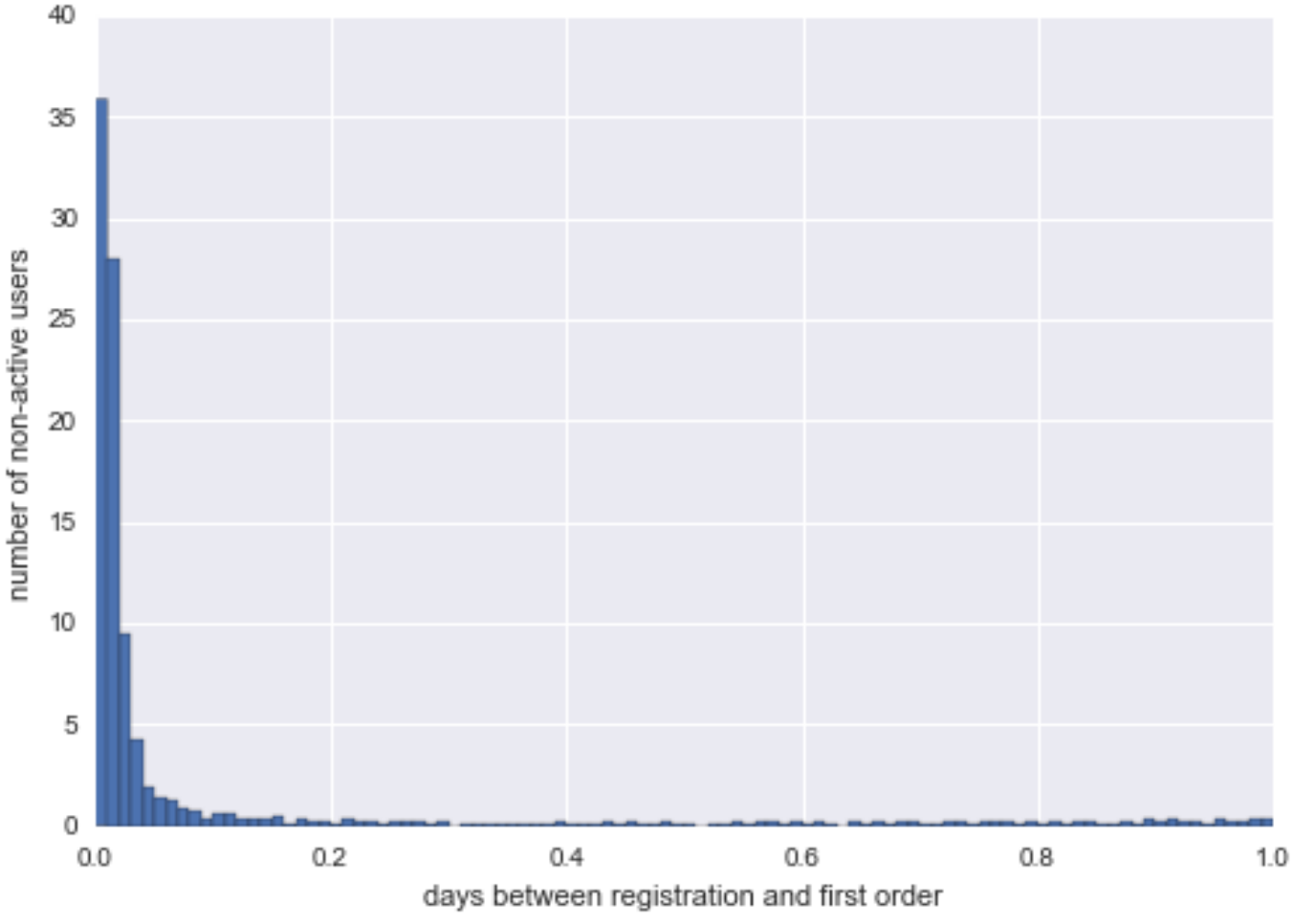
Remark: `sum_till_today` is a sum of all expenditures made until now

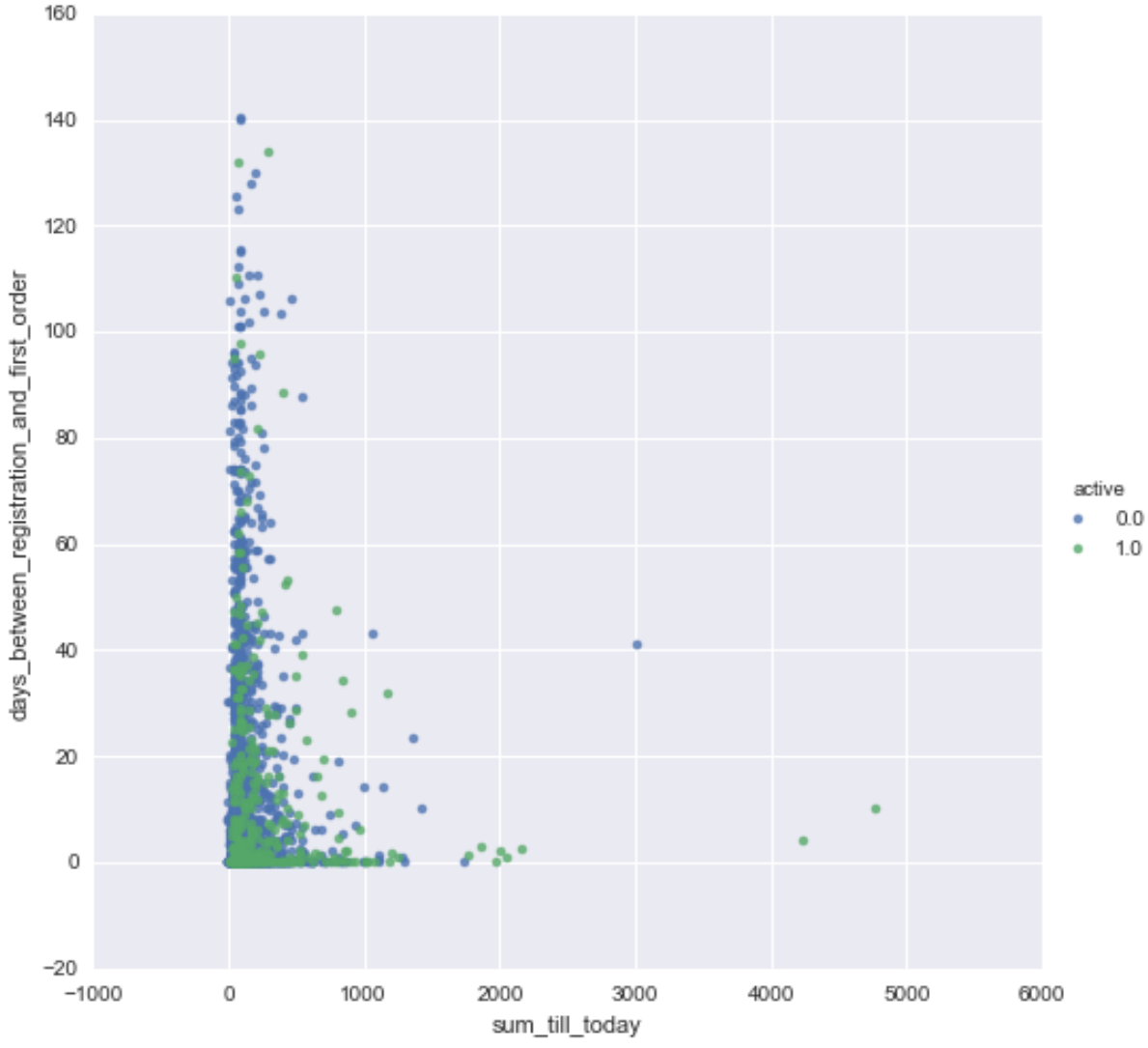


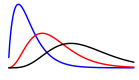



active
non-active
number of users
days between registration and first order
0
2
4
6
8
10
12
14
0
1
2
3
4
5


35
30
25
20
15
10
5
0
number of active users
0.0
0.2
0.4
0.6
0.8
1.0
days between registration and first order

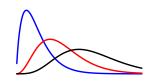

number of non-active users
days between registration and first order
0
5
10
15
20
25
30
35
40
0.0
0.2
0.4
0.6
0.8
1.0


160
140
120
100
80
60
40
20
0
−20
days_between_registration_and_first_order
−1000
0
1000
2000
3000
4000
5000
6000
sum_till_today
active
0.0
1.0



Test/train split

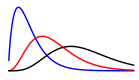
- 67% of users were used to estimate coefficients of the model, and 33% of users were used to calculate accuracy of prediction. One can use such a notation for this: test/train = 0.33



Predictive models

Notion of **confusion matrix** has been used in order to assess performance of the prediction model

By definition, a confusion matrix C is such that each element C_{ij} is equal to the number of observations known to be in group i but predicted to be in group j. Thus, confusion matrix simply shows how many obseravtions have been correctly classified(misclassified).

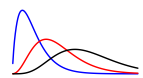


Naive Classifier

Idea of the Naive Classifier: calculate percentage of non-active users in the training set, and consider this number as estimated probability of being non-active. This probability of being non-active (**p.non.active**) can be used to classify users from the test set in such a way:

- consider a user from the test set, this user is non-active with probability **p.non.active**
- consider another user from the test set, this user is non-active with probability **p.non.active**
- ...

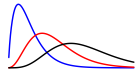
Naive Classifier can be considered as a simple benchmark to be compared with other predictive models



Naive Classifier in practice

Having applied **Naive Classifier** to predict user activity such **confusion matrix** has been obtained:

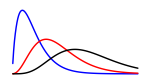
$$\begin{bmatrix} 879 & 120 \\ 105 & 14 \end{bmatrix}$$



Naive Classifier in practice

From the confusion matrix above it can be concluded that:

- number of non-active users is $879 + 120 = 999$
- number of non-active users classified as non-active is 879
- number of non-active users classified as active is 120
- number of active users is $105 + 14 = 119$
- number of active users classified as active is 14
- number of active users classified as non-active is 105
- percentage of correct classifications of non-active users: 88%
- percentage of correct classifications of active users: 12%
- percentage of correct classifications overall: 80%



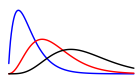
Support Vector Machine

Having applied **Support Vector Machine** to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 713 & 286 \\ 47 & 72 \end{bmatrix}$$

From the confusion matrix above it can be concluded that:

- percentage of correct classifications of non-active users: 71%
- percentage of correct classifications of active users: 61%
- percentage of correct classifications overall: 70%



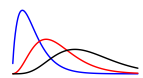
Support Vector Machine

By adjusting parameters of **Support Vector Machine** one can obtain higher percentage of correct classifications of active users, but it will lead to a lower percentage of correct classifications of non-active users. For instance, having applied **Support Vector Machine** with adjusted parameters to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 429 & 570 \\ 21 & 98 \end{bmatrix}$$

From the confusion matrix above it can be concluded that:

- percentage of correct classifications of non-active users: 43%
- percentage of correct classifications of active users: 82%



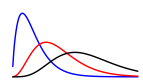
All models listed in the next slides perform better than **Naive Classifier**, but worse than **Support Vector Machine**.

Having applied **k-Nearest Neighbors algorithm** to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 903 & 96 \\ 82 & 37 \end{bmatrix}$$

From the confusion matrix above it can be concluded that:

- percentage of correct classifications of non-active users: 90%
- percentage of correct classifications of active users: 31%
- percentage of correct classifications overall: 84%



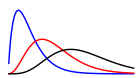
Linear Discriminant Analysis

Having applied **Linear Discriminant Analysis** to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 981 & 18 \\ 101 & 18 \end{bmatrix}$$

From the confusion matrix above it can be concluded that:

- percentage of correct classifications of non-active users: 98%
- percentage of correct classifications of active users: 15%
- percentage of correct classifications overall: 89%



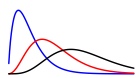
Quadratic Discriminant Analysis

Having applied **Quadratic Discriminant Analysis** to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 951 & 48 \\ 97 & 22 \end{bmatrix}$$

From the confusion matrix above it can be concluded that:

- percentage of correct classifications of non-active users: 95%
- percentage of correct classifications of active users: 18%
- percentage of correct classifications overall: 87%



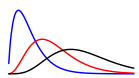
Decision Trees

Having applied **Decision Trees** to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 870 & 124 \\ 99 & 29 \end{bmatrix}$$

From the confusion matrix above it can be concluded that:

- percentage of correct classifications of non-active users: 87%
- percentage of correct classifications of active users: 24%
- percentage of correct classifications overall: 80%



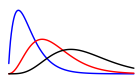
Random Forest

Having applied **Random Forest** to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 881 & 113 \\ 106 & 22 \end{bmatrix}$$

From the confusion matrix above it can be concluded that:

- percentage of correct classifications of non-active users: 88%
- percentage of correct classifications of active users: 18%
- percentage of correct classifications overall: 81%



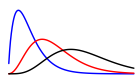
Naive Bayes

Having applied **Naive Bayes** to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 941 & 58 \\ 86 & 33 \end{bmatrix}$$

From the confusion matrix above it can be concluded that:

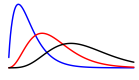
- percentage of correct classifications of non-active users: 94%
- percentage of correct classifications of active users: 28%
- percentage of correct classifications overall: 87%



New regressors in the model !

such features have been added(later **tracking and email data**):

- **sent 1 month**: number of emails sent to user within the last month
- **open 1 month**: number of emails opened by user within the last month
- **click 1 month**: number of emails clicked by user within the last month
- **session count 1 month**: number of sessions within the last month



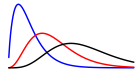
Support Vector Machine in action

Having applied **Support Vector Machine** to predict user activity such **confusion matrix** has been obtained:

$$\begin{bmatrix} 790 & 209 \\ 45 & 74 \end{bmatrix}$$

Confusion matrix of SVM before (without **tracking and email data**):

$$\begin{bmatrix} 713 & 286 \\ 47 & 72 \end{bmatrix}$$



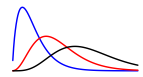
Support Vector Machine, test/train = 0.33

Confusion matrix (normalized) of SVM now:

$$\begin{bmatrix} 0.8058 & 0.1942 \\ 0.3931 & 0.6069 \end{bmatrix}$$

i.e. percentage of correct classif. of non-active users is 80.58%
Confusion matrix of SVM before:

$$\begin{bmatrix} 0.7137 & 0.2863 \\ 0.395 & 0.605 \end{bmatrix}$$



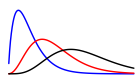
Support Vector Machine, test/train = 0.43

Confusion matrix of SVM now:

$$\begin{bmatrix} 0.8074 & 0.1926 \\ 0.3636 & 0.6364 \end{bmatrix}$$

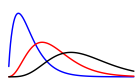
Confusion matrix of SVM before:

$$\begin{bmatrix} 0.716 & 0.284 \\ 0.4156 & 0.5844 \end{bmatrix}$$

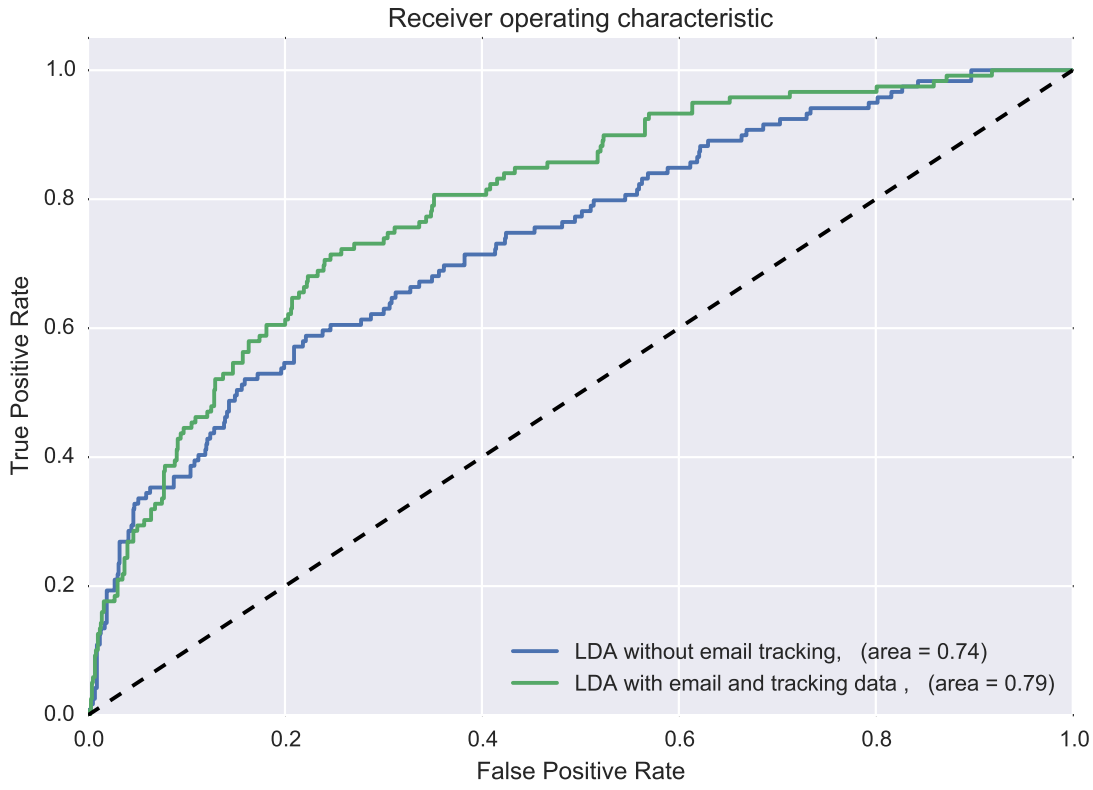


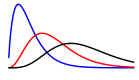
ROC curves

Receiver operating characteristic (ROC), or **ROC curve**, is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. So, ROC curve is a dynamic version of confusion matrix.

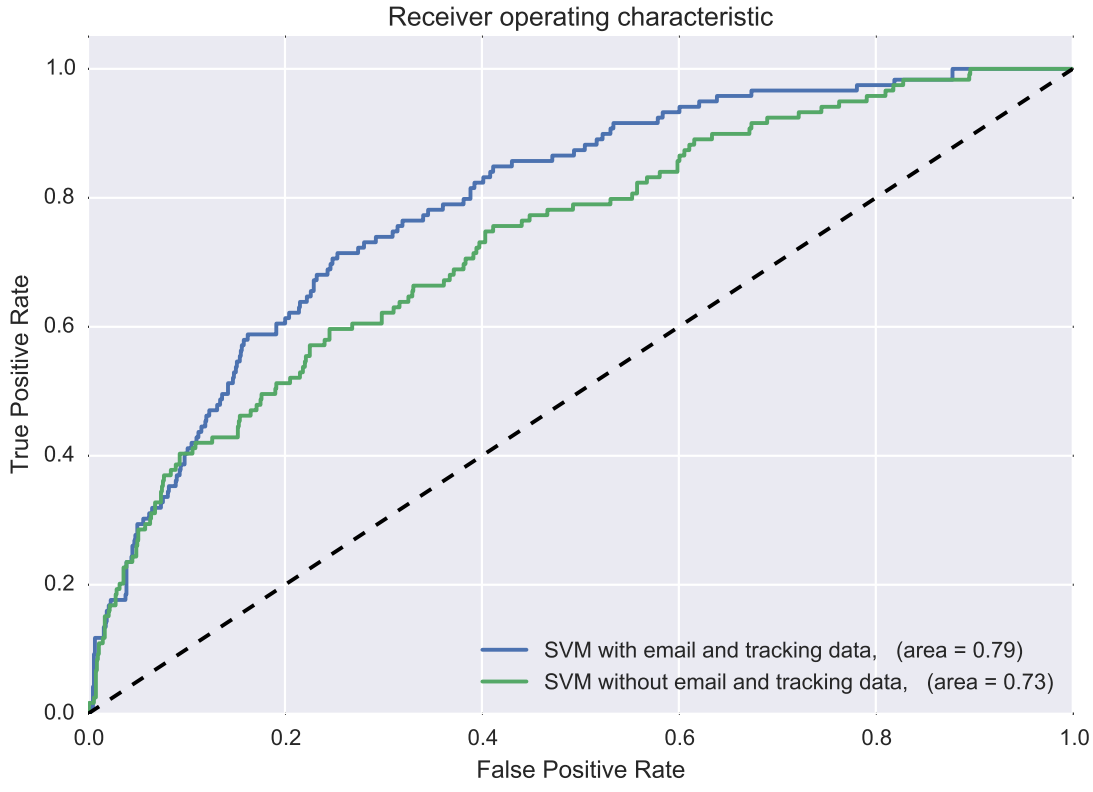


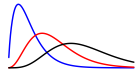
Linear Discriminant Analysis, ROC curve





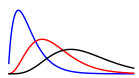
SVM (kernel: linear, weight: auto), ROC curve

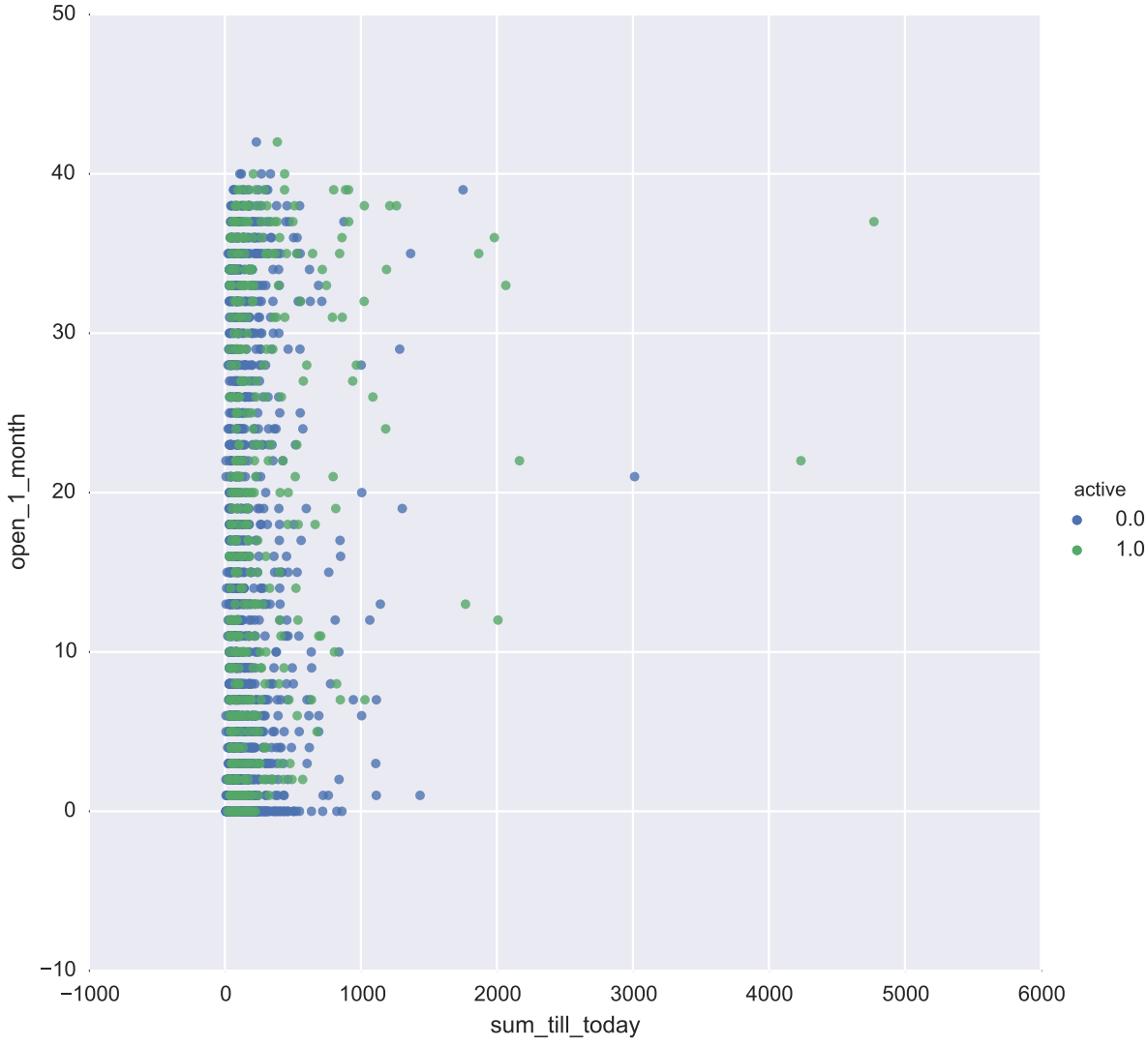


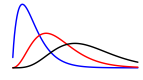


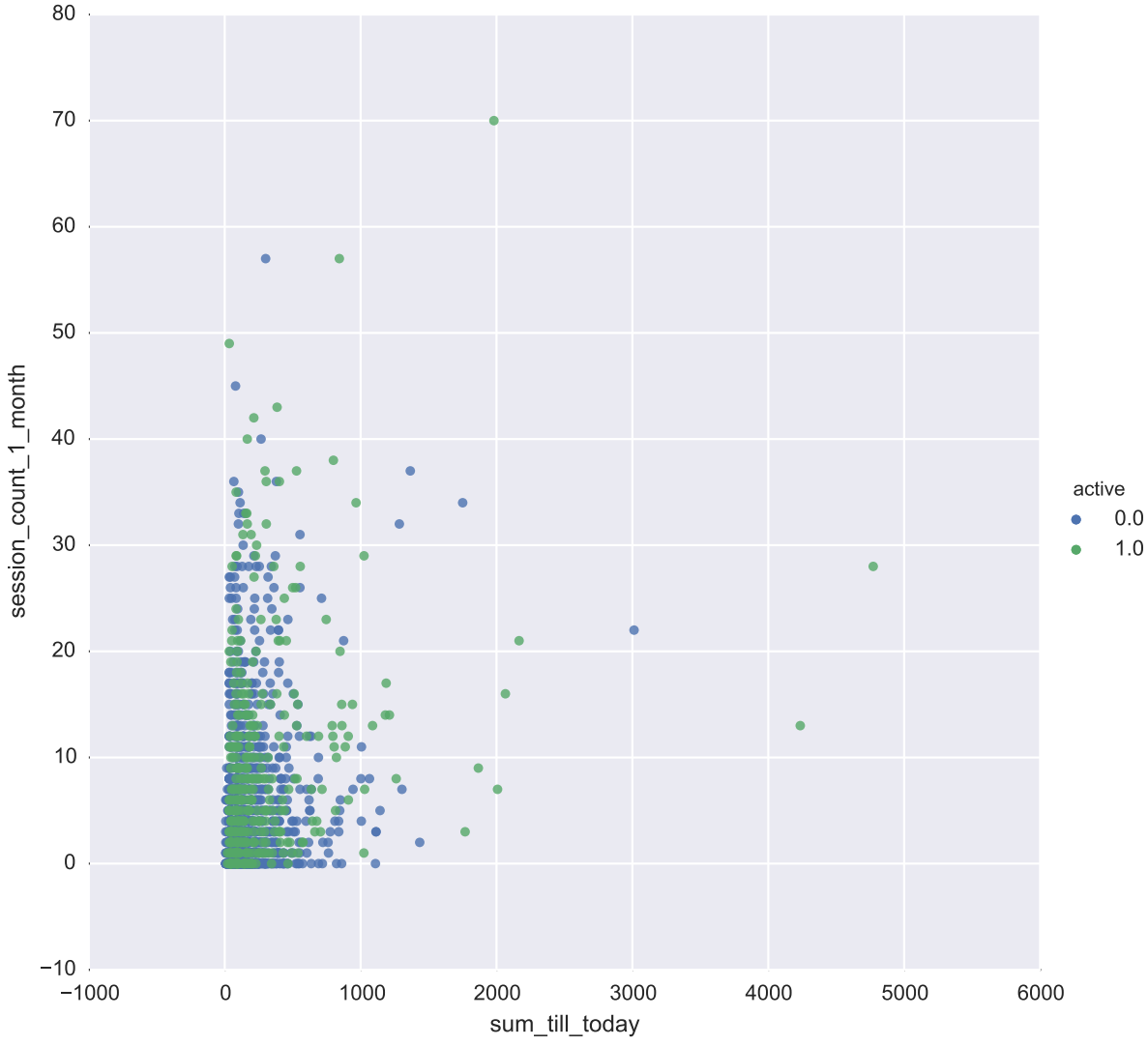
Intermediate results

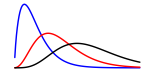
1. Two measures of the quality of the model:
   - **Confusion matrix**
   - **ROC curves**
2. The best performing model: **SVM**
3. Introduction of new regressors based on **tracking and email data** has improved quality of prediction (see ROC curves and confusion matrices).

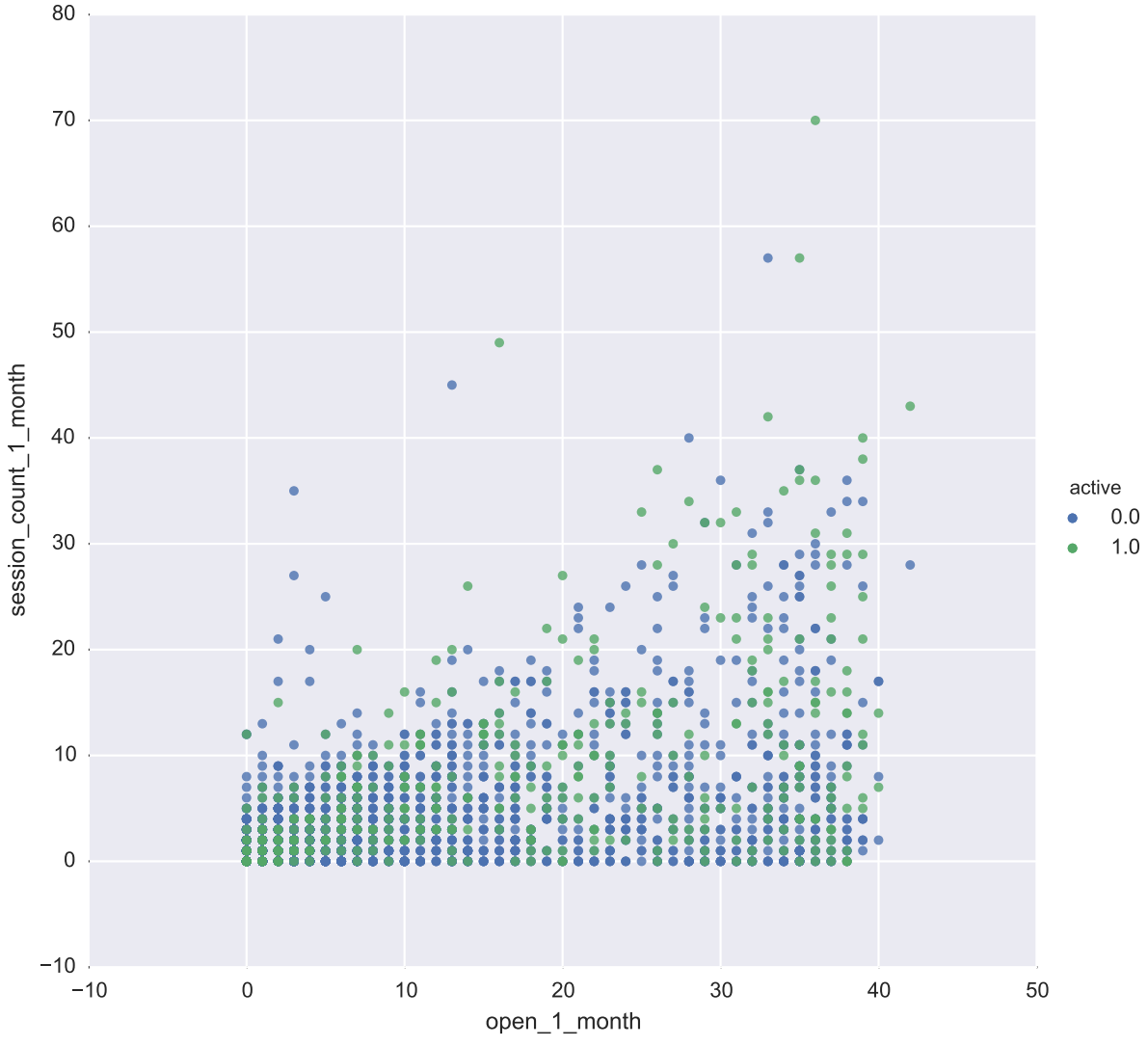


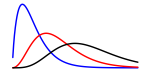

50
40
30
20
10
0
−10
open_1_month
−1000
0
1000
2000
3000
4000
5000
6000
sum_till_today
active
0.0
1.0




80
70
60
50
40
30
20
10
0
−10
session_count_1_month
−1000
0
1000
2000
3000
4000
5000
6000
sum_till_today
active
0.0
1.0



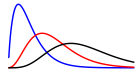

session_count_1_month
open_1_month
active
0.0
1.0
80
70
60
50
40
30
20
10
0
−10
−10
0
10
20
30
40
50



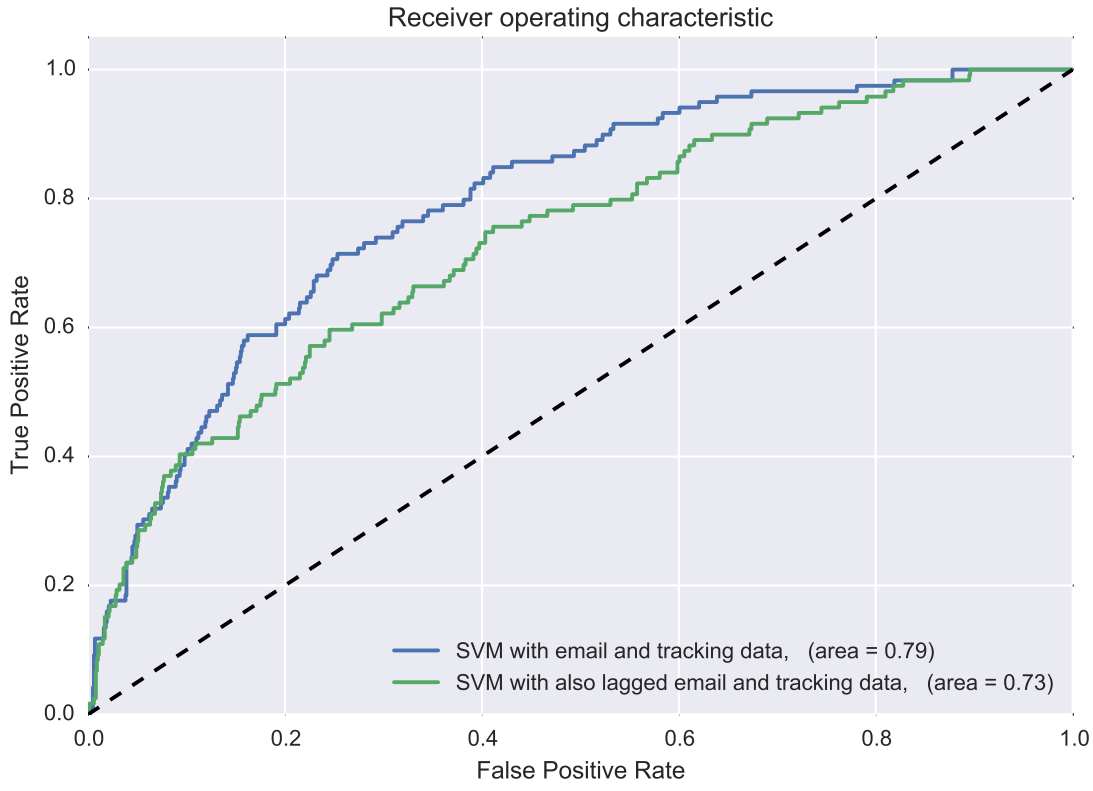
New regressors are lagged old regressors

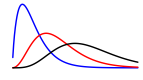
such features have been added(later **lagged tracking and email data**):

- **sent 2 month**: number of emails sent to user 2 months ago
- **open 2 month**: number of emails opened by user 2 months ago
- **click 2 month**: number of emails clicked by user 2 months ago
- **session count 2 month**: number of sessions of a user 2 months ago



SVM (kernel: linear, weight: auto)





Summary

Several models have been considered to predict user activity.

Support Vector Machine appeared to be the best performing model.

Having applied **Support Vector Machine** one obtains such an accuracy:

- percentage of correct classifications of non-active users:80.58%
- percentage of correct classifications of active users:60.50%