SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

Detecting Phishing Attempts with SAS®: Minimally Invasive Email Log Data

Taylor B. Anderson, Denise J. McManus

#SASGF

Detecting Phishing Attempts with SAS®: Minimally Invasive Email Log Data

Taylor B. Anderson, Denise J. McManus

THE UNIVERSITY OF ALABAMA® | Culverhouse College of Commerce

ABSTRACT

Phishing is the attempt of a malicious entity to acquire personal, financial, or otherwise sensitive information such as user names and passwords from recipients through the transmission of seemingly legitimate emails. By quickly alerting recipients of known phishing attacks, an organization can reduce the likelihood that a user will succumb to the request and unknowingly provide sensitive information to attackers. Methods to detect phishing attacks typically require the body of each email to be analyzed. However, most academic institutions do not have the resources to scan individual emails as they are received, nor do they wish to retain and analyze message body data. Many institutions simply rely on the education and participation of recipients within their network. Recipients are encouraged to alert information security (IS) personnel of potential attacks as they are delivered to their mailboxes. This poster explores a novel and more automated approach that uses SAS® to examine email header and transmission data to determine likely phishing attempts that can be further analyzed by IS personnel. Previously a collection of 2,703 emails from an external filtering appliance were examined with moderate success. This paper focuses on the gains from analyzing an additional 50,000 emails, with the inclusion of an additional 30 known attacks. Real-time email traffic is exported from Splunk Enterprise into SAS® for analysis. The resulting model aids in determining the effectiveness of alerting IS personnel to potential phishing attempts faster than a user simply forwarding a suspicious email to IS personnel.

MOTIVATION

Phishing attacks are a constant threat to any organization. End user education can help prevent users from being phished, but this is not always effective. Many organizations rely on users to forward suspicious emails to trained personnel for analysis, but again, this is not always performed by users. Once an email is identified, an announcement can be posted to recipients of the email so they are aware that the email should be deleted and, if they did succumb to the attack, the user should change their passwords and contact their organization's information security (IS) personnel. The motivation behind this work is to alert IS personnel to subject lines of potential phishing emails, thus, taking a proactive approach to detecting and eliminating suspicious emails. By using minimally invasive email log data, organizations that either do not or cannot analyze message bodies may be able to detect these attempts much quicker than relying on user notification. By detecting these emails earlier, users that have received the emails can be notified sooner (ideally before they even open the phishing email) and block rules can be added to the filtering appliance to prevent further delivery of the phishing emails.

GOALS

- Create a better SAS® regression model that integrates with Splunk Enterprise to alert IS personnel to likely phishing attempts
- Implement a basic threat score and database

OBTAINING THE DATA/METHODS

A python program was used to export 52,703 records from an email security appliance into CSV format

- 2,758 records from over 60 known phishing attacks
- 49,945 records were randomly selected from valid email traffic and spam email traffic

The python program was then used to remove unnecessary data and clean up missing fields

- The exported time format was converted to a SAS® supported format
- Empty message sizes were set to zero
- Empty spam status' were set to negative
- A new field called "has_attachment" was created with "1" denoting that an email had an attachment
- The attachment columns were dropped

The data was then imported into SAS® and further manipulated

- New variables were added to denote
    - Whether senders were internal faculty or staff, internal students, or external
    - Whether or not the message subject was UTF encoded
    - Whether the email appliance thought the email was spam
    - The number of recipients
- Final variables: has_attachment, hour_of_day, message_size, message_subject_contains_utf, number_of_recipients, sender_is_facstaff, sender_is_student, esa_spam_status
- SAS® Enterprise miner was used to create a logistic regression model to predict whether or not an email is likely to be part of a phishing attack. Other modeling techniques were considered previously and rejected. This poster explores the refinement of the logistic regression model. Significant factors in an email being a phishing email:
    - The subject line is not UTF encoded
    - The sender is not an internal user
    - The email has not been marked as spam
    - There is a high number of recipients
    - The message does not have an attachment
    - The message is small
    - The message was sent outside of work hours

The code was then exported from SAS® Enterprise Miner and used in base SAS®

Detecting Phishing Attempts with SAS®: Minimally Invasive Email Log Data

Taylor B. Anderson, Denise J. McManus

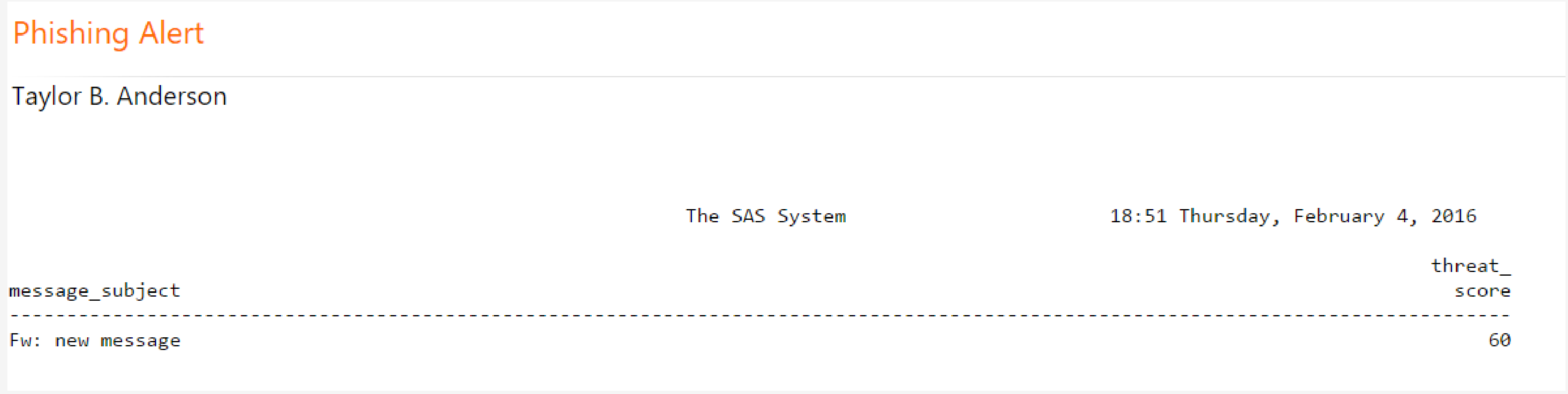
THE UNIVERSITY OF ALABAMA | Culverhouse College of Commerce

MODEL USED ON LIVE DATA

Live email traffic is fed into this resulting model using the same Python and SAS® programs above with some additional changes

- Five minutes of email traffic is captured every two minutes, resulting in a three minute overlap of the data.
- Any message subjects determined to not be part of a phishing attack by the model are removed.
- Message subjects with a probability of being part of a phishing attack less than 0.8 are removed.
- Messages sent to less than 5 people at a time are removed.
- A basic threat score is created using MAX_P=the max probability of all messages with a given subject line and COUNT=number of messages with a given subject line that were sent: THREAT_SCORE=MINIMUM(100,ROUND(10*MAX_P*COUNT))
- The resulting subject lines are compared to those within a database of past detected subject lines. New subject lines are added to the database and emailed to IS personnel for examination. Old subject lines are removed and the database last_seen field for that subject line is updated to the current timestamp.

MODEL USED ON LIVE DATA

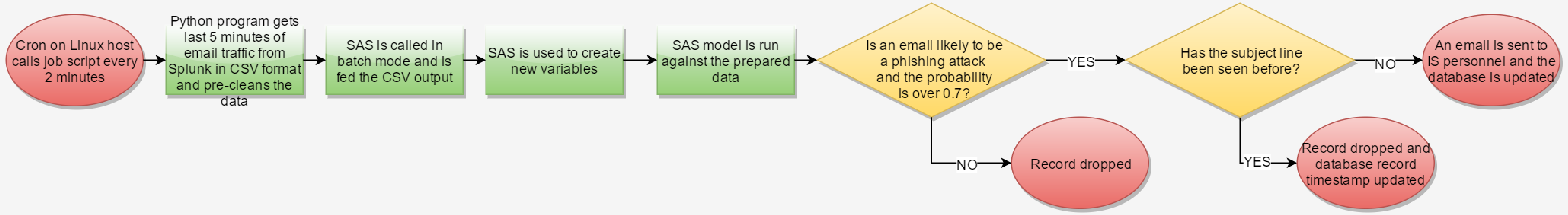

Phishing Alert

Taylor B. Anderson

```
                                              The SAS System                      18:51 Thursday, February 4, 2016

                                                                                                              threat_
message_subject                                                                                                 score
-------------------------------------------------------------------------------------------------------------------
Fw: new message                                                                                                    60
```

Email output from a small phishing attempt. Only 12 emails were delivered. Thanks to this successful test, IS personnel were able to send out an alert targeted to the recipients warning them to delete the email.


Cron on Linux host calls job script every 2 minutes
Python program gets last 5 minutes of email traffic from Splunk in CSV format and pre-cleans the data
SAS is called in batch mode and is fed the CSV output
SAS is used to create new variables
SAS model is run against the prepared data
Is an email likely to be a phishing attack and the probability is over 0.7?
YES
NO
Record dropped
Has the subject line been seen before?
NO
An email is sent to IS personnel and the database is updated
YES
Record dropped and database record timestamp updated

Detecting Phishing Attempts with SAS®: Minimally Invasive Email Log Data

Taylor B. Anderson, Denise J. McManus

THE UNIVERSITY OF ALABAMA® | Culverhouse College of Commerce

MODEL USED ON LIVE DATA

BASH Script

```
export PYTHONPATH=/opt/splunk-sdk-python; cd /opt/reg
/opt/reg/search.py --username=xxxxx --password=‘xxxxx’--host=xxxx --app=search 'search index=xxxx | transaction internal_message_id | table _time, recipient, sender, message_size, message_subject, spam_status, attachment_type, file_name' --earliest_time="$(date +"%Y-%m-%dT%T.000" --date '-3 min')" --latest_time="$(date +"%Y-%m-%dT%T.000")" --output_mode=csv > temp.csv
# replace _time with datetime and clean up datetime format
/usr/bin/sed -i 's/_time/datetime/g' temp.csv
/usr/bin/sed -i 's/\....-0/-0/g' temp.csv
# do additional row cleanup
/usr/bin/python pre_sas_cleanup.py >> temp2.csv
/usr/bin/mv temp2.csv temp.csv
# Call SAS
/usr/local/SASHome/SASFoundation/9.4/bin/sas_en –nodms /opt/reg/new_esa_regression.sas
/usr/bin/cat *.lst | egrep -v "SPAM|Undeliverable" > emailbody.tmp
if [ -s emailbody.tmp ];
then
# REPLACED* – send the email and add the lines to the database
fi
/usr/bin/rm *.lst emailbody.tmp temp.csv 2> /dev/null
/usr/bin/echo "$(/usr/bin/date) - ran" >> log.txt
```

*Small portion of the bash script that runs the entire process. Email and database portions were removed to protect the organization.

RESULTS

Training:

Misclassification rate = 36%

False Negative = 11%

True Negative = 16%

False Positive = 25%

True Positive = 48%

Validation:

Misclassification rate = 35%

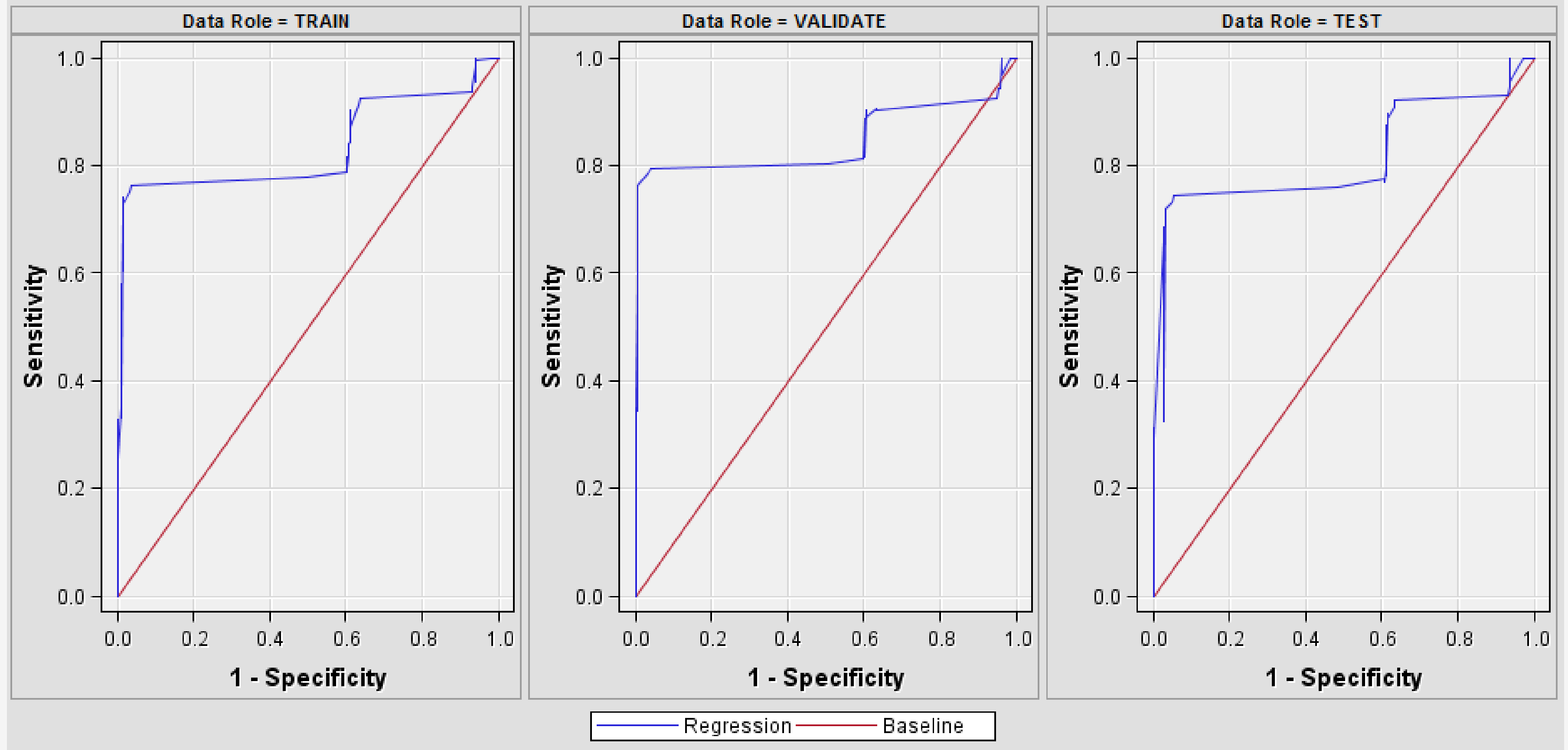
False Negative = 10%

True Negative = 16%

False Positive = 25%

True Positive = 49%

In any case, false positives are acceptable. This revision of the model better classifies phishing emails than the previous model and also classifies less official announcements. This is due to the inclusion of the hour_of_day bin variable and split from the previous "is_internal_domain" variable into faculty/staff and student domains. However, this model also incorrectly classifies more spam and advertising email than the previous. This will be remedied in future analysis of subject line text.



FUTURE WORK

While this effort significantly improved on the number of misclassifications, particularly false positives, from the original regression model, it still requires future work. Text analysis of the message subjects should further reduce the number of mass mailers and spam emails that are passed through the model as false negatives while improving true positive results by catching incorrect grammar, spelling, and other errors. This will also require that UTF and ISO encoded subject lines are decoded before performing text analysis.

SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

LAS VEGAS | APRIL 18-21

#SASGF