# Text Analytics:
# Spam Classification

Team 1-03: Eric Fung, Michael McKenna, Rebecca Wood

# Agenda

- **Business Problem**
- **Data Information**
- **Models**
  - Naive-Bayes
  - Generalized Linear Model
  - Generalized Additive Model
  - Support Vector Machines
    - *Linear*
    - *Radial*
    - *Polynomial*
  - Classification Trees
    - *Information*
    - *Gini*
  - Random Forest
    - *Boosting*
- **Conclusion**
- **Further Research**

# Business Problem

- With the recent increase in phishing, spoofing and email advertising, spam emails have made their way into customers' inboxes.
- Zoho, based out of California, has hired us to improve upon their spam detection model for their company-client based email service.

# UCI Machine Learning Repository Data

- ➢ 4601 emails (observations)
- ➢ 57 attributes (predictors)
  - ○ Most indicate whether a word or character is frequently occuring in an email
  - ○ Run length attributes measure frequency of consecutive capital letters
- ➢ 1 response variable
  - ○ Spam: 1
  - ○ Not spam: 0
- ➢ Emails broken down into 57 attributes:
  - ○ *Word frequency* - top 48 words
  - ○ *Character frequency* - special characters
  - ○ *Capitalization* - run length, total

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | v1 | v2 | v3 | v4 | v5 | v6 | v7 | v8 | v9 | v10 | v11 | v12 | v13 | v14 | v15 |
| 2 | 0 | 0.64 | 0.64 | 0 | 0.32 | 0 | 0 | 0 | 0 | 0 | 0 | 0.64 | 0 | 0 | 0 |
| 3 | 0.21 | 0.28 | 0.5 | 0 | 0.14 | 0.28 | 0.21 | 0.07 | 0 | 0.94 | 0.21 | 0.79 | 0.65 | 0.21 | 0.1 |
| 4 | 0.06 | 0 | 0.71 | 0 | 1.23 | 0.19 | 0.19 | 0.12 | 0.64 | 0.25 | 0.38 | 0.45 | 0.12 | 0 | 1.7 |
| 5 | 0 | 0 | 0 | 0 | 0.63 | 0 | 0.31 | 0.63 | 0.31 | 0.63 | 0.31 | 0.31 | 0.31 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0.63 | 0 | 0.31 | 0.63 | 0.31 | 0.63 | 0.31 | 0.31 | 0.31 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 1.85 | 0 | 0 | 1.85 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 1.92 | 0 | 0 | 0 | 0 | 0.64 | 0.96 | 1.28 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 1.88 | 0 | 0 | 1.88 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0.15 | 0 | 0.46 | 0 | 0.61 | 0 | 0.3 | 0 | 0.92 | 0.76 | 0.76 | 0.92 | 0 | 0 | 0 |
| 11 | 0.06 | 0.12 | 0.77 | 0 | 0.19 | 0.32 | 0.38 | 0 | 0.06 | 0 | 0 | 0.64 | 0.25 | 0 | 0.1 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0.96 | 0 | 0 | 1.92 | 0.96 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0.25 | 0 | 0.38 | 0.25 | 0.25 | 0 | 0 | 0 | 0.12 | 0.12 | 0.12 | 0 | 0 |
| 14 | 0 | 0.69 | 0.34 | 0 | 0.34 | 0 | 0 | 0 | 0 | 0 | 0 | 0.69 | 0 | 0 | 0 |
| 15 | 0 | 0 | 0 | 0 | 0.9 | 0 | 0.9 | 0 | 0 | 0.9 | 0.9 | 0 | 0.9 | 0 | 0 |
| 16 | 0 | 0 | 1.42 | 0 | 0.71 | 0.35 | 0 | 0.35 | 0 | 0.71 | 0 | 0.35 | 0 | 0 | 0 |
| 17 | 0 | 0.42 | 0.42 | 0 | 1.27 | 0 | 0.42 | 0 | 0 | 1.27 | 0 | 0 | 0 | 0 | 0 |
| 18 | 0 | 0 | 0 | 0 | 0.94 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20 | 0 | 0 | 0.55 | 0 | 1.11 | 0 | 0.18 | 0 | 0 | 0 | 0 | 0 | 0.92 | 0 | 0.1 |
| 21 | 0 | 0.63 | 0 | 0 | 1.59 | 0.31 | 0 | 0 | 0.31 | 0 | 0 | 0.63 | 0 | 0 | 1.2 |
| 22 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 23 | 0.05 | 0.07 | 0.1 | 0 | 0.76 | 0.05 | 0.15 | 0.02 | 0.55 | 0 | 0.1 | 0.47 | 0.02 | 0 | 0 |
| 24 | 0 | 0 | 0 | 0 | 2.94 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 25 | 0 | 0 | 0 | 0 | 1.16 | 0 | 0 | 0 | 0 | 0 | 0 | 0.58 | 0 | 0 | 0 |
| 26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 27 | 0.05 | 0.07 | 0.1 | 0 | 0.76 | 0.05 | 0.15 | 0.02 | 0.55 | 0 | 0.1 | 0.47 | 0.02 | 0 | 0 |
| 28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 29 | 0 | 0 | 0 | 0 | 0 | 0 | 1.66 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

spambase

# Naive-Bayes Model

**Model:**

- `Method = 'nb'`
- Repeated 10-fold cross-validation
- Data is centered and scaled
- Processed predictors with principal component analysis (PCA)

**Results:**

- 87 % accuracy
- Type 1 Error: 15 %
- Type 2 Error: 11 %

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 472  39
         1  85 323

               Accuracy : 0.8651
                 95% CI : (0.8413, 0.8865)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7236
 Mcnemar's Test P-Value : 5.32e-05

            Sensitivity : 0.8474
            Specificity : 0.8923
         Pos Pred Value : 0.9237
         Neg Pred Value : 0.7917
             Prevalence : 0.6061
         Detection Rate : 0.5136
   Detection Prevalence : 0.5560
      Balanced Accuracy : 0.8698

       'Positive' Class : 0
```

# Generalized Linear Model (GLM)

**Model:**

- ➢ `Method = 'bayesglm'`
- ➢ Repeated 10-fold cross-validation
- ➢ Data is centered and scaled
- ➢ Processed predictors with principal component analysis (PCA)

**Results:**

- ➢ 92 % accuracy
- ➢ Type 1 Error: 5 %
- ➢ Type 2 Error: 12 %

```
Confusion Matrix and Statistics

          Reference
Prediction 0   1
         0 529 42
         1 28  320

               Accuracy: 0.9238
                 95% CI: (0.9047, 0.9401)
    No Information Rate: 0.6061
    P-Value [Acc > NIR]: <2e-16

                  Kappa: 0.8394
 Mcnemar's Test P-Value: 0.1202

            Sensitivity: 0.9497
            Specificity: 0.8840
         Pos Pred Value: 0.9264
         Neg Pred Value: 0.9195
             Prevalence: 0.6061
         Detection Rate: 0.5756
   Detection Prevalence: 0.6213
      Balanced Accuracy: 0.9169

       'Positive' Class : 0
```

# Generalized Additive Model (GAM)

**Model:**

- `Method = 'gamboost'`
- Repeated 10-fold cross-validation
- Data is centered and scaled
- Processed predictors with principal component analysis (PCA)

**Results:**

- 92 % accuracy
- Type 1 Error:  5 %
- Type 2 Error:  12 %

```
Confusion Matrix and Statistics

          Reference
Prediction  0   1
         0 529  45
         1  28 317

               Accuracy : 0.9206
                 95% CI : (0.9012, 0.9372)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : < 2e-16

                  Kappa : 0.8323
 Mcnemar's Test P-Value : 0.06112

            Sensitivity : 0.9497
            Specificity : 0.8757
         Pos Pred Value : 0.9216
         Neg Pred Value : 0.9188
             Prevalence : 0.6061
         Detection Rate : 0.5756
   Detection Prevalence : 0.6246
      Balanced Accuracy : 0.9127

       'Positive' Class : 0
```

# Support Vector Machine – Linear

**Model:**

- ➢ Repeated 10-fold cross-validation
- ➢ Data is centered and scaled
- ➢ Processed predictors with principal component analysis (PCA)
- ➢ Tuned for best cost value

**Results:**

- ➢ 93 % accuracy
- ➢ Type 1 Error:  5 %
- ➢ Type 2 Error:  12 %

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 531  45
         1  26 317

               Accuracy : 0.9227
                 95% CI : (0.9035, 0.9392)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : < 2e-16

                  Kappa : 0.8367
 Mcnemar's Test P-Value : 0.03266

            Sensitivity : 0.9533
            Specificity : 0.8757
         Pos Pred Value : 0.9219
         Neg Pred Value : 0.9242
             Prevalence : 0.6061
         Detection Rate : 0.5778
   Detection Prevalence : 0.6268
      Balanced Accuracy : 0.9145

       'Positive' Class : 0
```

# Support Vector Machine – Radial

**Model:**

- ➢ Repeated 10-fold cross-validation
- ➢ Data is centered and scaled
- ➢ Processed predictors with principal component analysis (PCA)
- ➢ Tuned for best cost and sigma values

**Results:**

- ➢ 93 % accuracy
- ➢ Type 1 Error: 5 %
- ➢ Type 2 Error: 10 %

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 530  37
         1  27 325

               Accuracy : 0.9304
                 95% CI : (0.9119, 0.946)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8534
 Mcnemar's Test P-Value : 0.2606

            Sensitivity : 0.9515
            Specificity : 0.8978
         Pos Pred Value : 0.9347
         Neg Pred Value : 0.9233
             Prevalence : 0.6061
         Detection Rate : 0.5767
   Detection Prevalence : 0.6170
      Balanced Accuracy : 0.9247

       'Positive' Class : 0
```

# Support Vector Machine - Polynomial

**Model:**

- ➢ Data is centered and scaled
- ➢ Repeated 10-fold cross-validation
- ➢ Processed predictors with PCA
- ➢ Tuned for best cost, scale, and degree values

**Results:**

- ➢ 93 % accuracy
- ➢ Type 1 Error: 4 %
- ➢ Type 2 Error: 11 %

```
Confusion Matrix and Statistics

          Reference
Prediction 0   1
         0 534 39
         1 23 323

               Accuracy: 0.9325
                 95% CI: (0.9143, 0.9479)
    No Information Rate: 0.6061
    P-Value [Acc > NIR]: < 2e-16

                  Kappa: 0.8576
 Mcnemar's Test P-Value: 0.05678

            Sensitivity: 0.9587
            Specificity: 0.8923
         Pos Pred Value: 0.9319
         Neg Pred Value: 0.9335
             Prevalence: 0.6061
         Detection Rate: 0.5811
   Detection Prevalence: 0.6235
      Balanced Accuracy: 0.9255

       'Positive' Class : 0
```

# Classification Trees – Information

**Model:**

- ➢ Create initial tree
- ➢ Identify optimal `cp`, `minbucket`, `minsplit` parameters
- ➢ Prune tree and evaluate test set

**Results:**

- ➢ 92 % accuracy
- ➢ Type 1 Error:  6 %
- ➢ Type 2 Error:  10 %

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 521  35
         1  36 327

               Accuracy : 0.9227
                 95% CI : (0.9035, 0.9392)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8383
 Mcnemar's Test P-Value : 1

            Sensitivity : 0.9354
            Specificity : 0.9033
         Pos Pred Value : 0.9371
         Neg Pred Value : 0.9008
             Prevalence : 0.6061
         Detection Rate : 0.5669
   Detection Prevalence : 0.6050
      Balanced Accuracy : 0.9193

       'Positive' Class : 0
```

# Classification Trees – Gini

**Model:**

- ➢ Enumerate through values of `minsplit` & `minbucket`
- ➢ Identify `cp`, `minbucket`, `minsplit` parameters with highest accuracy
- ➢ Prune tree using this information
- ➢ Evaluate test set performance

**Results:**

- ➢ 92 % accuracy
- ➢ Type 1 Error: 6 %
- ➢ Type 2 Error: 11 %

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 525  41
         1  32 321

               Accuracy : 0.9206
                 95% CI : (0.9012, 0.9372)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8329
 Mcnemar's Test P-Value : 0.3491

            Sensitivity : 0.9425
            Specificity : 0.8867
         Pos Pred Value : 0.9276
         Neg Pred Value : 0.9093
             Prevalence : 0.6061
         Detection Rate : 0.5713
   Detection Prevalence : 0.6159
      Balanced Accuracy : 0.9146

       'Positive' Class : 0
```

# Random Forest

**Model:**

- ➢ Find optimal `mtry` value
  - ○ Enumerate through vector of values
- ➢ Train new random forest model
- ➢ Predict on test set
- ➢ Compute accuracy scores

**Results:**

- ➢ 95 % accuracy
- ➢ Type 1 Error:  3 %
- ➢ Type 2 Error:  8 %

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 525  41
         1  32 321

               Accuracy : 0.9206
                 95% CI : (0.9012, 0.9372)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8329
 Mcnemar's Test P-Value : 0.3491

            Sensitivity : 0.9425
            Specificity : 0.8867
         Pos Pred Value : 0.9276
         Neg Pred Value : 0.9093
             Prevalence : 0.6061
         Detection Rate : 0.5713
   Detection Prevalence : 0.6159
      Balanced Accuracy : 0.9146

       'Positive' Class : 0
```

# Random Forest – AdaBoost

**Model:**

- ➢ Find optimal `nu` value
- ➢ Rebuild boost tree with new `nu` value
- ➢ Train new boost model
- ➢ Predict on test set
- ➢ Compute accuracy scores

**Results:**

- ➢ 96 % accuracy
- ➢ Type 1 Error:  3 %
- ➢ Type 2 Error:  6 %

```
Confusion Matrix and Statistics

          Reference
Prediction 0   1
         0 538  21
         1  19 341

               Accuracy : 0.9565
                 95% CI : (0.9412, 0.9687)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.9088
 Mcnemar's Test P-Value : 0.8744

            Sensitivity : 0.9659
            Specificity : 0.9420
         Pos Pred Value : 0.9624
         Neg Pred Value : 0.9472
             Prevalence : 0.6061
         Detection Rate : 0.5854
   Detection Prevalence : 0.6083
      Balanced Accuracy : 0.9539

       'Positive' Class : 0
```

# Random Forest – GBM Boost

**Model:**

- ➢ Find optimal parameters
- ➢ Train boost model with optimal shrinkage and decision boundary
- ➢ Predict on test set
- ➢ Compute accuracy scores

**Results:**

- ➢ 94 % accuracy
- ➢ Type 1 Error:  4 %
- ➢ Type 2 Error:  8 %

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 533  30
         1  24 332

               Accuracy : 0.9412
                 95% CI : (0.924, 0.9556)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8766
 Mcnemar's Test P-Value : 0.4962

            Sensitivity : 0.9569
            Specificity : 0.9171
         Pos Pred Value : 0.9467
         Neg Pred Value : 0.9326
             Prevalence : 0.6061
         Detection Rate : 0.5800
   Detection Prevalence : 0.6126
      Balanced Accuracy : 0.9370

       'Positive' Class : 0
```

# Random Forest - XGBoost

**Model:**

- ➢ Train initial XGBoost model
- ➢ Identify optimal decision boundary
- ➢ Predict on test set

**Results:**

- ➢ 96 % accuracy
- ➢ Type 1 Error:  4 %
- ➢ Type 2 Error:  5 %

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 533  17
         1  24 345

               Accuracy : 0.9554
                 95% CI : (0.94, 0.9678)
    No Information Rate : 0.6061
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.9069
 Mcnemar's Test P-Value : 0.3487

            Sensitivity : 0.9569
            Specificity : 0.9530
         Pos Pred Value : 0.9691
         Neg Pred Value : 0.9350
             Prevalence : 0.6061
         Detection Rate : 0.5800
   Detection Prevalence : 0.5985
      Balanced Accuracy : 0.9550

       'Positive' Class : 0
```

# Conclusion

| | Accuracy | Type1 | Type2 |
|---|---|---|---|
| NaiveBayes | 86.50707 | 15.26032 | 10.773481 |
| GLM | 92.38303 | 5.026930 | 11.602210 |
| GAM | 92.05658 | 5.026930 | 12.430939 |
| SVMLinear | 92.27421 | 4.667864 | 12.430939 |
| SVMRadial | 93.03591 | 4.847397 | 10.220994 |
| SVMPolynomial | 93.25354 | 4.129264 | 10.773481 |
| TreeInformation | 92.27421 | 6.463196 | 9.6685080 |
| TreeGini | 92.05658 | 5.745063 | 11.325967 |
| RandomForest | 95.10337 | 2.692998 | 8.2872930 |
| AdaBoost | 95.64744 | 3.411131 | 5.8011050 |
| GbmBoost | 94.12405 | 4.308797 | 8.2872930 |
| XgBoost | 95.53863 | 4.308797 | 4.6961330 |

**Optimal Model: XGBoost**

# Next Steps

- Accounting for Type 2 Errors: willing to sacrifice accuracy for lower false positives
  - Adding or removing attributes
  - Using decision boundaries to identify Type 2 Errors
  - Use ROC curve to reduce Type 2 Errors
- Use hard-coded rules for outliers and remaining emails
- Aggregate various models