SOCIOL 229 Mini-Project 2:
Classifying Spam Email with Kernelized SVMs

Ezra Hsieh

June 9, 2025

1 High-Level Description of the Substantive Problem

The proliferation of unsolicited commercial email, commonly known as spam, presents a significant challenge for internet users and service providers. Effective automated filtering is crucial for maintaining the usability of email as a communication tool. This project addresses the substantive problem of creating an accurate spam filter by applying a kernelized Support Vector Machine (kSVM), a powerful machine learning technique, to classify emails as either spam or legitimate (ham).

2 The Data Set

This analysis utilizes the Spambase Data Set from the UCI Machine Learning Repository [1]. The data were collected at Hewlett-Packard Labs in 1999 and consist of 4,601 email messages, of which 1,813 (39.4%) are spam. The dataset is designed for binary classification tasks.

The data are described by 58 attributes:

- **48 continuous attributes** representing the frequency of specific words (e.g., "free", "business", "george").
- **6 continuous attributes** representing the frequency of specific characters (e.g., '!', '$', '#').
- **3 continuous attributes** measuring the run-lengths of consecutive capital letters (average, longest, and total).
- **1 nominal class attribute** indicating whether the email is spam (1) or not spam (0).

As noted by the data creators, the non-spam emails were collected from personal and work accounts, meaning terms like "george" and "hpl" are strong indicators of legitimate email in this specific dataset. An 80/20 split was used to create a training set of 3,680 instances and a hold-out test set of 921 instances for final model evaluation.

3 Description of the Learning Problem

3.1 Substantive Terms

The primary goal is to build a model that can accurately distinguish between unsolicited commercial spam and legitimate personal or professional emails. Such a model should be able to identify patterns in language, character usage, and formatting that are characteristic of each class, providing a reliable automated filter to reduce inbox clutter.

3.2 Formal Terms

In formal terms, this is a supervised binary classification problem.

- **Outcome Variable (y):** A binary label where $y = 1$ if an email is spam and $y = -1$ if it is not spam.
- **Predictors ($\mathbf{x}$):** A feature vector of 57 continuous attributes representing word/character frequencies and capital letter statistics.
- **Model Objective:** To learn a decision function $f(\mathbf{x})$ that correctly predicts the class label of a new, unseen email. Specifically, we use a kernelized Support Vector Machine (kSVM) to find an optimal separating hyperplane in a high-dimensional feature space.

4 Analysis Strategy

The chosen method for this project is the kernelized Support Vector Machine (kSVM), as implemented in the `kernelTools` R package [2]. This approach was selected because of its effectiveness in high-dimensional spaces and its ability to find non-linear decision boundaries, both of which are highly relevant for text-based data like email content. SVMs aim to find a maximum-margin hyperplane, which often leads to strong generalization performance.

The analysis strategy involved the following steps:

1. **Data Partitioning:** The data was split into an 80% training set and a 20% test set to ensure an unbiased evaluation of the final model's performance.
2. **Hyperparameter Tuning:** To find the optimal model specification, 5-fold cross-validation was performed on the training data using the `kernSVMBW` function. This process was repeated for four different kernel functions to explore a range of possible decision boundaries:
   - **Linear:** To provide a baseline performance measure.
   - **Polynomial:** To capture polynomial relationships.
   - **Radial Basis Function (RBF):** A flexible kernel capable of mapping data into an infinite-dimensional space, effective for complex class boundaries.
   - **Arc-Cosine:** A kernel well-suited for high-dimensional data.

   For each kernel, a grid search was conducted over its specific hyperparameters (e.g., 'sigma' for RBF, 'degree' for polynomial) and a common regularization parameter, 'C' (from 10^{-3} to 10^{3}).
3. **Final Model Selection and Evaluation:** The kernel and hyperparameter combination yielding the highest cross-validated accuracy was selected. This final model was then trained on the entire training dataset and evaluated on the held-out test set.
4. **Interpretation:** The final model was interpreted by examining the out-of-sample confusion matrix, calculating standard performance metrics, and analyzing the correlation between the raw predictors and the model's decision function to identify key features.

5 Results

5.1 Kernel and Hyperparameter Tuning

The performance of the four kernel types was tuned using 5-fold cross-validation on the training data. Table 1 summarizes the best cross-validated accuracy achieved for each kernel. The Radial Basis Function (RBF) kernel with a regularization parameter of $C = 10$ and $\sigma = 0.1$ achieved the highest cross-validated accuracy of 81.22%. Based on this result, the RBF kernel was selected for the final model.

Table 1: Best 5-Fold Cross-Validation Accuracy by Kernel Type

Kernel Type	**Best Reg. Param (C)**	**Best Kern. Param**	**CV Accuracy**
RBF	10.0	$\sigma = 0.1$	**0.8122**
Arc-Cosine	0.01	order = 1	0.7654
Linear	0.01	None	0.7574
Polynomial	0.001	degree = 1	0.7457

5.2 Final Model Performance

The selected RBF model was trained on the full training set and evaluated on the held-out test set. The model achieved a high in-sample accuracy of 96.41%, but the more realistic out-of-sample accuracy was 81.00%. The out-of-sample confusion matrix is shown in Figure 1 and detailed performance metrics are provided in Table 2.

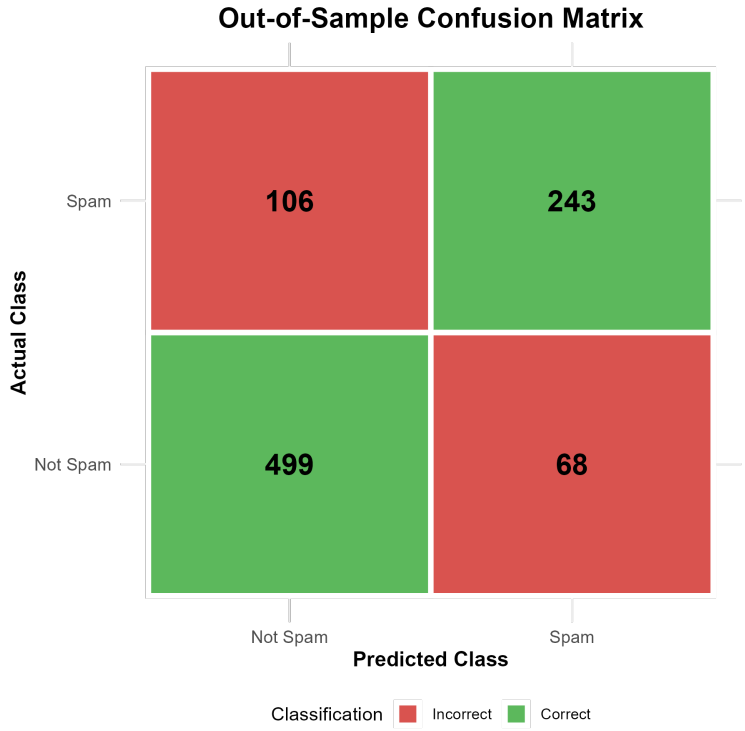

Figure 1: Out-of-Sample Confusion Matrix for the Final kSVM Model.

The model demonstrates strong performance, correctly classifying 499 non-spam and 243 spam emails. The high specificity (TNR) of 0.8801 indicates it is particularly good at correctly identifying legitimate emails, while the recall of 0.6963 suggests it misses a higher proportion of actual spam emails.

Table 2: Final Model Performance Metrics on Test Data

Metric	**Value**
Accuracy	0.8100
Precision (PPV)	0.7814
Recall (Sensitivity)	0.6963
F1-Score	0.7364
Specificity (TNR)	0.8801

5.3 Feature Interpretation

To understand which features were most influential, we examine the correlation between each feature and the model's raw prediction score. A positive correlation suggests the feature is indicative of spam, while a negative correlation suggests it is indicative of non-spam.

Table 3 shows that the presence of dollar signs and words like "your," "all," and "order" are strongly associated with spam. Longer sequences of capital letters also serve as a key indicator.

Table 3: Top 10 Features Indicating 'Spam'

Feature	**Correlation**
word_freq_your	0.2490
word_freq_you	0.2352
capital_run_length_total	0.1899
word_freq_all	0.1743
word_freq_our	0.1625
char_freq_$	0.1542
word_freq_000	0.1538
word_freq_will	0.1471
word_freq_order	0.1421
word_freq_business	0.1404

Conversely, Table 4 confirms the dataset documentation's note that words specific to the creators' context, such as "george", "hpl", and various area codes, are the strongest predictors of legitimate email. This highlights the personalized nature of the trained filter.

Table 4: Top 10 Features Indicating 'Not Spam' (Ham)

Feature	Correlation
word_freq_george	-0.5329
word_freq_address	-0.1641
word_freq_857	-0.0101
word_freq_415	-0.0088
word_freq_telnet	-0.0001
word_freq_table	0.0035
word_freq_direct	0.0045
word_freq_lab	0.0078
word_freq_project	0.0090
word_freq_cs	0.0105

6 Conclusions

This project successfully developed an effective spam filter using a kernelized Support Vector Machine with a Radial Basis Function (RBF) kernel. The final model achieved an out-of-sample accuracy of 81.0%, demonstrating its ability to generalize to new, unseen data.

The analysis of feature correlations provides substantive insights into the model's decision-making process. The model correctly identified that spam is characterized by commercial language, generic references (money and dollar signs), and aggressive formatting (long capital letter runs). Likewise, it learned that legitimate emails in this dataset were characterized by personal names and work-related terms. This confirms that the kSVM approach can effectively capture the complex, non-linear patterns present in email text to perform accurate classification.

7 Next Steps

While the current model is successful, several avenues exist for future work.

- **Addressing Data-Specific Biases:** The strong negative correlation with terms like "george" and "hpl" highlights that the model is highly personalized. For a more general-purpose filter, these features might need to be removed or the model would need to be retrained on a more diverse corpus of non-spam emails.
- **Optimizing for False Positives:** The documentation notes that false positives (classifying ham as spam) are particularly undesirable. Although the model has a high true negative rate (specificity) of 0.8801, it could be further optimized to reduce false positives.
- **Other Kernels:** Exploring compositions of kernels or more complex models in classification accuracy.

References

[1] Mark Hopkins, Erik Reeber, George Forman, and Jaap Suermondt. *Spambase Data Set*. UCI Machine Learning Repository, 1999. `https://doi.org/10.24432/C53G6X`.

[2] Butts, Carter T. *kernelTools: Select Tools for Kernel Learning*. R package version 0.8, 2023.