**SMS Spam Classification Report**

**Introduction**  
This report compares three approaches for classifying SMS messages as “ham” or “spam” using a bag-of-words representation (each feature counts words appearing in ≥1% of messages). We evaluate a random-guess baseline, logistic regression, and two deep neural network (DNN) variants. Performance is measured by accuracy, precision, recall, F1 Score, and ROC AUC.

**Data and Methods**  
The dataset contains 1,574 messages: 1,366 ham (87%) and 208 spam (13%). Features V2–V1261 record word counts; V1 is the label. We split data 80/20 maintaining the rate of spam calls after the split

**The three methods used are:**

* **Random baseline:** predict spam with probability = training spam rate.
* **Logistic regression:** single sigmoid layer trained with binary cross-entropy (Keras).
* **DNN:** Dense(64) → Dropout(0.5) → Dense(32) → Dropout(0.5) → Sigmoid.

**Performance**  
The random baseline achieved 75.9% accuracy, 13.4% recall, and 12.5% precision on spam Logistic regression rose to 98.6% accuracy, 89.3% recall, 100% precision, and 0.983 ROC AUC

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | **accuracy** | **precision** | **recall** | **f1\_score** | **roc\_auc** |
| Random Guessing | 0.7587 | 0.1250 | 0.1342 | 0.1295 | 0.5000 |
| Logistic | 0.9865 | 1.0000 | 0.8993 | 0.9470 | 0.9868 |
| Neural Network | 0.9857 | 0.9854 | 0.9060 | 0.9441 | 0.9793 |

Both models outperform random guessing, but the performance is surprisingly similar with comparable accuracy and F1\_scores. If this similar performance is in fact true, my inclination would be to use the logistic model over DNN.

**Diagnostic Plots**

A graph with blue and orange lines

AI-generated content may be incorrect.

A graph of a logistic curve

AI-generated content may be incorrect.

A screenshot of a computer screen

AI-generated content may be incorrect.A screenshot of a computer screen

AI-generated content may be incorrect.

**Discussion and Conclusion**  
Logistic regression delivers near‐optimal performance on this count‐based representation with minimal preprocessing. The DNN requires careful tuning—scaling, class weights, batch normalization, and dropout adjustments—to approach the linear model’s results. This complexity introduces greater computational overhead and risk of misconfiguration. Additionally, raw accuracy can be misleading: an always‐ham predictor would score 87% accuracy but fail to detect spam. Therefore, metrics such as recall, precision, F1 score, and ROC AUC are better for evaluating models on imbalanced data. In this context, a simple logistic model offers the best trade‑off between ease of use and reliable spam detection, while a DNN may be justified only if future features demand nonlinear modeling.

.