A Comparative Analysis Of Supervised Learning Algorithms

Manavi Shukla[1]

[1]*gtid:903952938, mshukla40@gatech.edu*

Abstract

This report delves into the performance of three supervised learning algorithms: Neural Networks, Support Vector Machines, and k-Nearest Neighbors. Two unique and interesting classification problems ad taken to investigates the behavior of these algorithms across different datasets. A thorough experimental approach was adopted, involving data exploration, hyper-parameter tuning, and the creation of validation/loss curves and confusion matrices to visualize results. Training and testing error rates were analyzed to validate hypotheses about each dataset's performance. The outcomes of this research highlight the comparative advantages and limitations of each algorithm, offering valuable insights for their application in diverse machine learning tasks.

1. Introduction

This study compares the performance of Neural Networks, Support Vector Machines, and k-Nearest Neighbors on two distinct classification problems: predicting diabetes and customer churn. Through Exploratory Data Analysis (EDA) and hypothesis formulation, we investigate the behavior of these algorithms on contrasting datasets, utilizing learning curves, cross validation scores, ROC/AUC, and confusion matrix to validate our hypotheses and offer valuable insights for selecting the most suitable algorithm for diverse machine learning tasks.

2. Dataset Understanding

The first data chosen for analysis is Diabetes Health Indicators Dataset[4] and the second one is Customer Churn Dataset.[5]

2.1. Data 1: Diabetes prediction

The Diabetes Health Indicators Dataset is an essential resource for understanding the predictive factors associated with diabetes. In clinical applications, both recall (sensitivity) and precision are critical, as they determine the model's effectiveness in correctly identifying diabetic patients without an excessive number of false positives, necessitating a robust F1 score for effective evaluation.

One significant challenge with this dataset is the presence of outliers. These anomalies reflect real-world occurrences and are clinically relevant, such as extremely high BMI values that may indicate severe health issues. Therefore, it is imperative to analyze how the inclusion of these outliers affects model performance.

Moreover, with 19 features available, the dataset provides an excellent opportunity to evaluate whether reducing dimensions by removing low-information features can enhance the model's performance. The class imbalance, with a disproportionately higher number of non-diabetic instances, poses another significant challenge. Addressing this imbalance through undersampling is crucial for developing a reliable classification model.

2.2. Data 2: Churn Prediction

The Churn Prediction Dataset provides a valuable platform for examining customer retention dynamics.

Characterized by a smaller number of features with clear separability, presents an interesting case for developing predictive models. Feature pair-plots reveal distinct clusters that facilitate model training. In this context, achieving high recall and precision for both churners and non-churners is essential, making the F1 score a suitable evaluation metric.

Although the dataset shows minor class imbalance, it still be addressed to avoid biased predictions. Furthermore, it includes slight skewness in some numerical features, which may affect model performance by introducing biases or reducing the effectiveness of certain algorithms. We aim to explore how these factors influence the predictive power of the implemented models and how they can be mitigated or leveraged to enhance performance.

3. Hypothesis

3.1. Diabetes Prediction

- Hypothesis 1: Identifying and retaining high-information features will lead to more accurate predictions.
- Hypothesis 2: The inclusion of outliers significantly impacts the performance of classification models. Using a Robust Scaler should mitigate their effect.
- Hypothesis 3: Undersampling the majority class points closest to the minority class will improve separability and model performance.
- Hypothesis 4: ANN will overperform over other machine learning models due to the complex relationship between the featuers and the target variables, which can be better captured by ANN.

3.2. Churn Prediction

- Hypothesis 1: Balancing the minority class through oversampling will improve model performance.
- Hypothesis 2: The RBF kernel will perform best for SVM due to non-linear feature separability.
- Hypothesis 3: KNN will outperform other models due to clear feature separation.

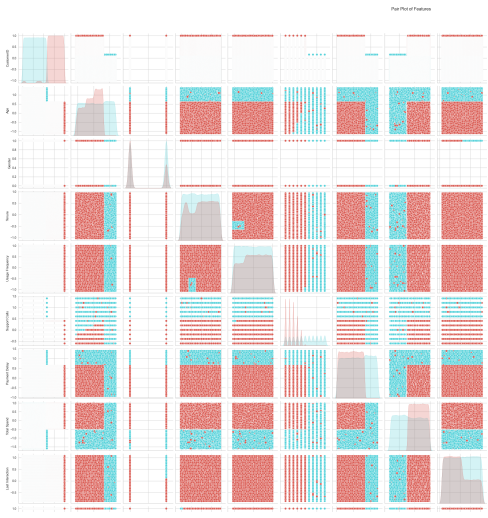

Figure 3.1. N-Feature paiplot of Churn Dataset

4. Experiment

4.1. Artificial Neural Network

ANNs are machine learning model which consist of interconnected nodes across structured layers that process information similar to human brain.

Tuning Depth and Width - Neurons and Layers, can improve an NN's ability to learn complex relationships within the data which

could be beneficial for both churn (capturing complex customer behavior patterns) and diabetes (modeling non-linear relationships between health factors). But it can lead to over-fitting, where the model memorizes the training data but performs poorly on unseen data.

4.1.1. Learning Rate

Finding the right learning rate is essential for both churn and diabetes data to ensure the model converges effectively and avoids getting stuck or over-fitting. Additionally, we are using early stopping at a tolerance of $10^{-}3$ to avoid over-fitting.

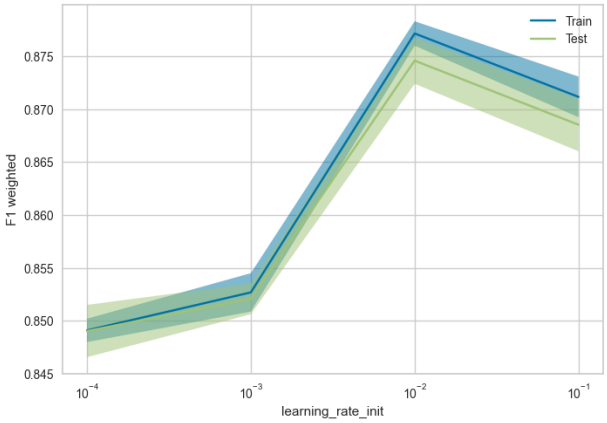

Figure 4.1. Validation curve for learning Rate, Dataset 1

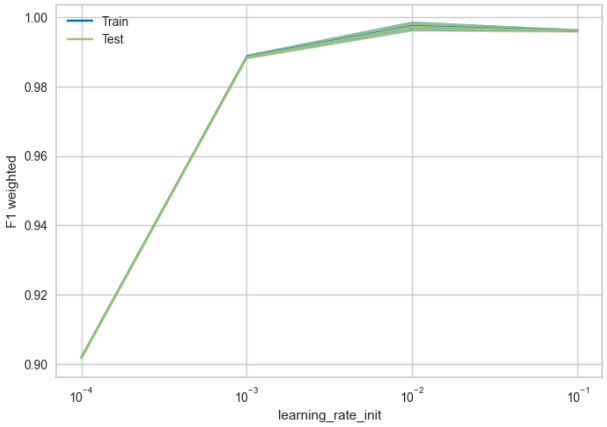

Figure 4.2. Validation curve for learning Rate, Dataset 2

4.1.2. Activation Function

The activation function plays a crucial role in artificial neural networks (ANNs) for several reasons-

- Activation functions introduce non-linearity into the ANN, allowing it to learn and model these intricate relationships between features and the target variable.
- Training an ANN involves back propagation, where we adjust the weights and biases of the network by feeding back the errors obtained in the outputs. Activation functions that have smooth gradients like Relu or tanh help in efficient gradient updates that are fed back through the network during training.[6]

Considering Activation Function Choices:[3]

1. ***Identity:*** *This linear function doesn't introduce non-linearity. It might be useful for the output layer in regression problems where the model directly predicts the target variable.*
2. ***Logistic (Sigmoid):*** *Outputs values between 0 and 1, making it suitable for classification problems as it predicts the probability of a class. However, sigmoid can suffer from vanishing gradients in deep networks.*
3. ***TanH:*** *The Hyperbolic Tan function outputs values between -1 and 1, offering a centered distribution around zero. This can mitigate the vanishing gradient problem better than the sigmoid function, although it is not immune to it entirely.*
4. ***ReLU (Rectified Linear Unit):*** *It is computational efficient and has the ability to avoid vanishing gradients. Outputs the input value if it's positive, otherwise outputs 0.*

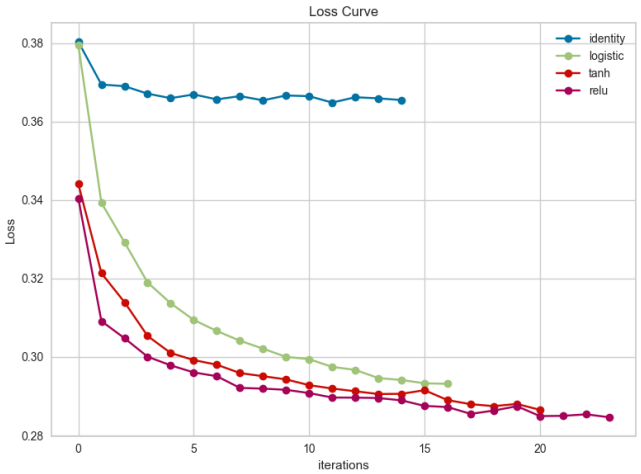

Figure 4.3. Loss Curve for different activation functions, Dataset1

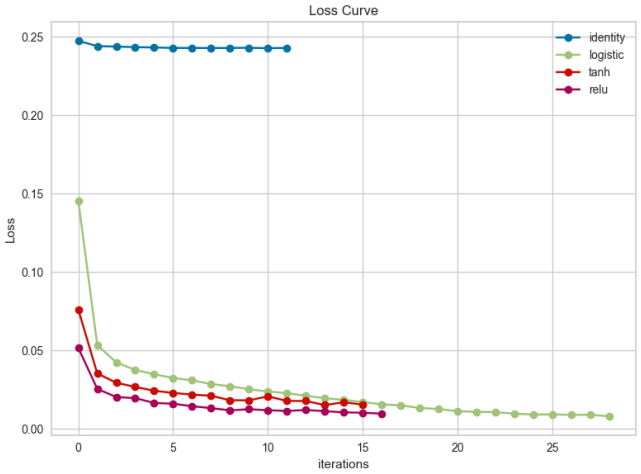

Figure 4.4. Loss Curve for different activation functions, Dataset2

The best voted parameter based on the weighted F1 mean score for both dataset is shown in Table 1.

Dataset	lr	Activation	Hidden Layer Size
Diabetes Data	0.01	Relu	(50, 50)
Churn Data 2	0.01	Relu	(50,)

Table 1. Best HyperParameters for each dataset using grid search

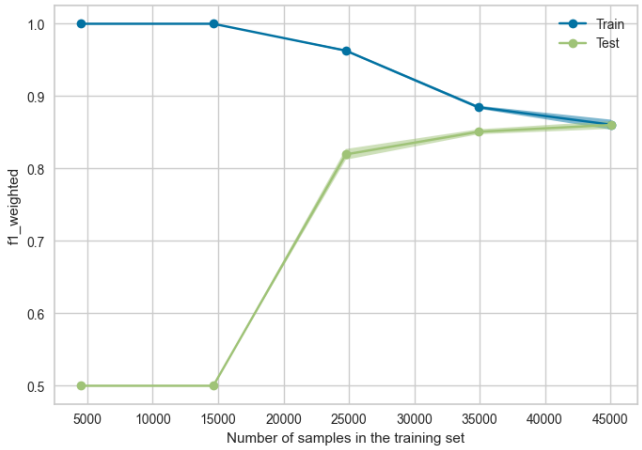

Figure 4.5. Learning Curve: Train/test samples vs F1 weighted score, Dataset1

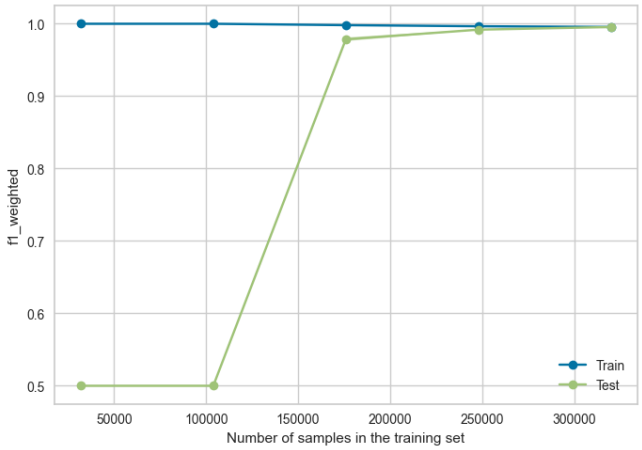

Figure 4.6. Learning Curve: Train/test samples vs F1 weighted score, Dataset2

4.2. Support Vector Machine

Support Vector Machines are particularly adept at identifying the optimal hyperplane that best separates data points from different classes with the greatest possible margin. A significant strength of SVMs is their capacity to manage nonlinear data through the implementation of kernel functions. These functions map the input data into a higher-dimensional space, making it possible to find a linear boundary that effectively segregates the classes in this transformed space. [2]

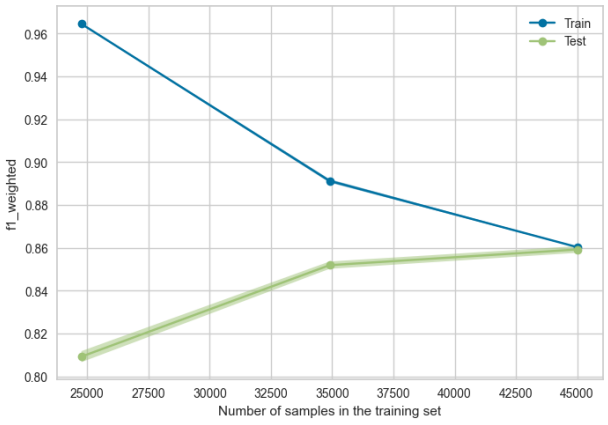

Figure 4.7. Train/Test Samples vs F1 Score, Dataset1

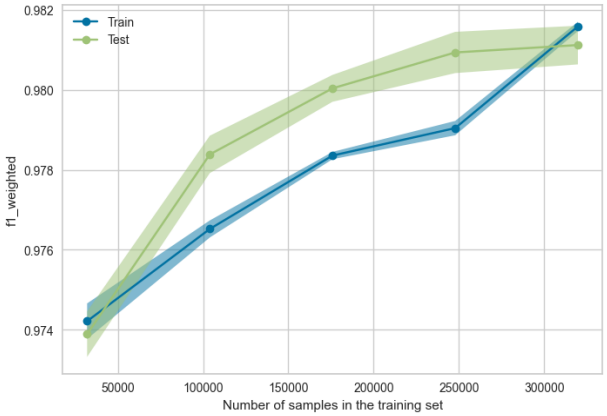

Figure 4.8. Train/Test Samples vs F1 Score, Dataset2

We compare the performance of following kernels on both datasets:

- Linear Kernel: Suitable for Linearly separable data. As per our feature pairplots, not all datapoints are linearly separable, hence this kernel will not be the most suitable.

$$K(x, y) = x \cdot y^T \tag{1}$$

- Radial Basis Function (RBF) Kernel: RBF is a good choice for both diabetes and churn data, as there is nonlinear relationships between features and the target variable. RBF also works well as a general-purpose kernel.

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right) \tag{2}$$

  RBF has a crucial parameter, gamma, that controls the width of the Gaussian function. A high gamma leads to overfitting.
- Polynomial Kernel: Potentially useful for capturing non-linear relationships between features, especially if those relationships can be expressed as polynomials. This kernel is not suitable for our datasets.

$$K(x, y) = (x \cdot y^T + c)^d \tag{3}$$

- Sigmoid Kernel: Similar to the polynomial kernel, it implicitly maps data to a higher-dimensional space, but the specific mapping is less interpretable. Not always as effective as other kernels like RBF in capturing complex non-linearities but can be computationally expensive.

$$K(x, y) = \tanh(x \cdot y^T + c) \tag{4}$$

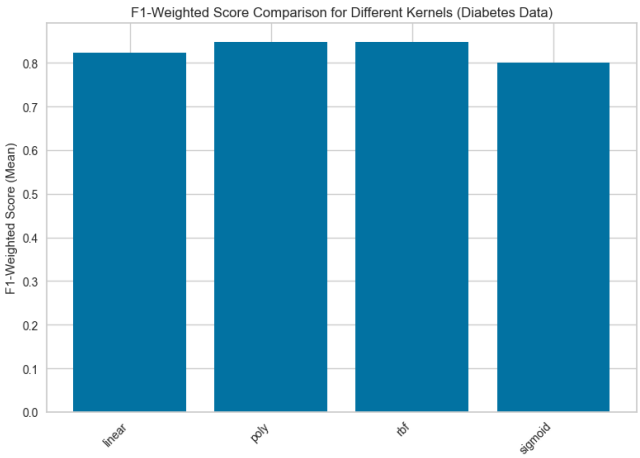

Figure 4.9. F1-weighted Score for different kernels, Dataset1

Dataset	Kernel	C	Gamma
Diabetes	rbf	1.0	1
Churn	rbf	0.01	1000

Table 2. Best parameter values for SVM

4.2.1. Other Parameters

- **C:** The regularization parameter that controls the trade-off between achieving a low error on the training data and minimizing the model complexity. A smaller value of C indicates a simpler model that may tolerate more errors on the training data but generalizes better. For the Churn dataset, C=0.1 suggests a preference for a simpler model, while C=1 for the Diabetes dataset indicates a slightly higher tolerance for training errors to achieve better fit.
- γ**:** The kernel coefficient for the RBF kernel, influencing the decision boundary. A high γ value means the decision boundary is influenced more by individual data points, potentially leading to overfitting. For the Churn dataset, a high γ=1000 indicates a very flexible decision boundary, whereas γ =1 for the Diabetes dataset suggests a more generalized decision boundary.

4.3. K Nearest Neighbours

K-Nearest Neighbors (KNN) is a non-parametric machine learning algorithm that classifies data points based on the predominant class among their k nearest neighbors. The parameter k, which represents the number of neighbors considered, is critical for tuning to enhance predictive performance. A smaller k value focuses on the closest neighbors, leading to more detailed and localized predictions, which can be beneficial for datasets with clear separations, such as churn prediction. Conversely, a larger k value incorporates more neighbors, resulting in a smoother decision boundary but potentially diminishing sensitivity to outliers. [1]

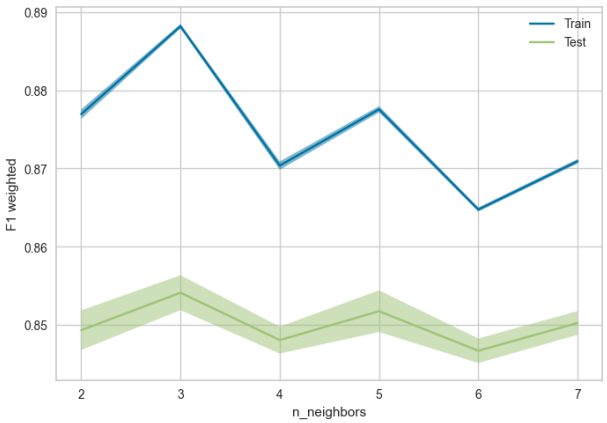

Figure 4.10. Validation Curve: K (Diabetes)

We see that due to the simple data, K neighbours are enough to vote the correct label for Dataset 1. Similar results were seen for Churn Dataset as well.

5. Results

5.1. Evaluation Metrics

To evaluate the performance of each of our models, the following evaluation metrics were taken into consideration.

1. **Accuracy:** The ratio of correctly predicted instances to the total instances. It indicates the overall effectiveness of the model but it can be misleading if the dataset like ours has class imbalance.
2. **Precision:** The ratio of true positive predictions to the total positive predictions (true positives + false positives). Higher precision indicates a low false positive rate
3. **Recall:** The ratio of true positive predictions to the total actual positives (true positives + false negatives). High recall indicates a low false negative rate. For both our Diabetes and Churn prediction - a High recall is a crucial.
4. **F1 Score:** The F1 score is the harmonic mean of precision and recall. It combines both metrics to give a single score that balances the trade-off between precision and recall. We have tuned our hyper-parameters based on the weighted F1 score. This is an important metric to capture.
5. **ROC AUC (Receiver Operating Characteristic Area Under the Curve):** Measures the area under the ROC curve, which plots the true positive rate against the false positive rate at various threshold settings. It gives combined metric of performance across all classification thresholds. It can be used to compare different models and underst the trade-offs between true positive and false positive rates.

5.2. Model Comparison on Dataset 1: Diabetes

Metric	ANN	SVM	KNN
Accuracy	0.852	0.859	0.855
Precision	0.857	0.865	0.866
Recall	0.852	0.859	0.855
F1-Score	0.852	0.858	0.853
ROC AUC	0.853	0.859	0.855

Table 3. Model Performance on Dataset 1: Diabetes

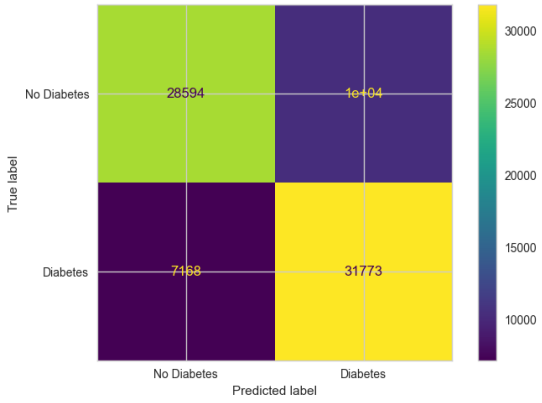

Figure 5.1. Dataset1 Confusion Matrix: ANN

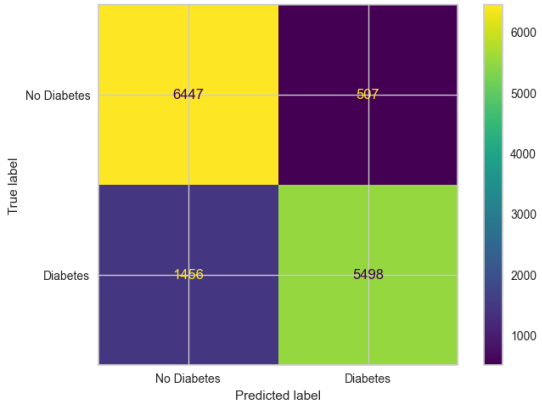

Figure 5.2. Dataset1 Confusion Matrix: SVM

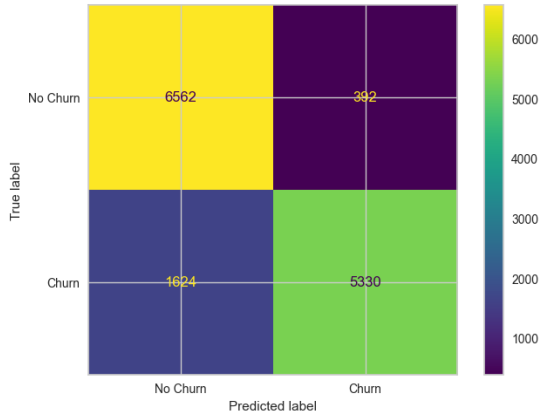

Figure 5.3. Dataset1 Confusion Matrix: KNN

5.3. Model Comparison on Dataset 2: Churn

Metric	ANN	SVM	KNN
Accuracy	0.99	0.781	0.962
Precision	0.99	0.828	0.965
Recall	0.99	0.781	0.962
F1-Score	0.99	0.773	0.962
ROC AUC	0.99	0.781	0.962

Table 4. Model Performance on Dataset 2: Churn

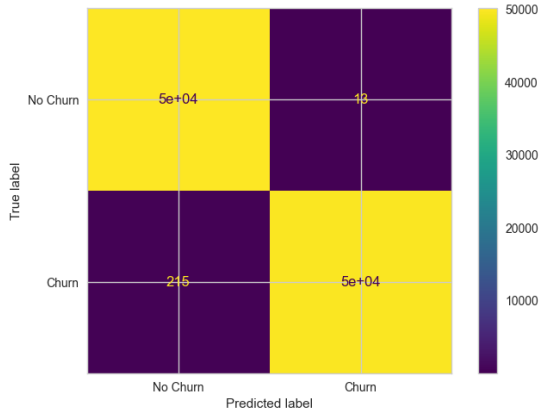

Figure 5.4. Dataset1 Confusion Matrix: ANN

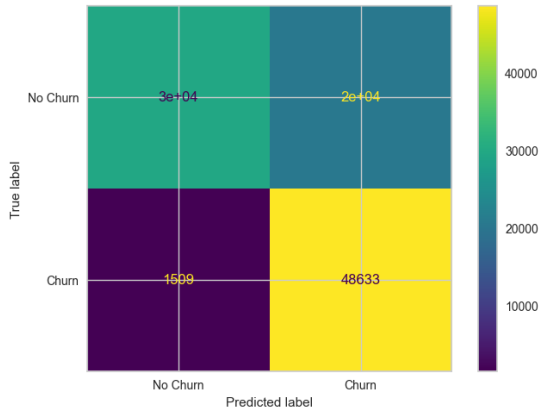

Figure 5.5. Dataset1 Confusion Matrix: SVM

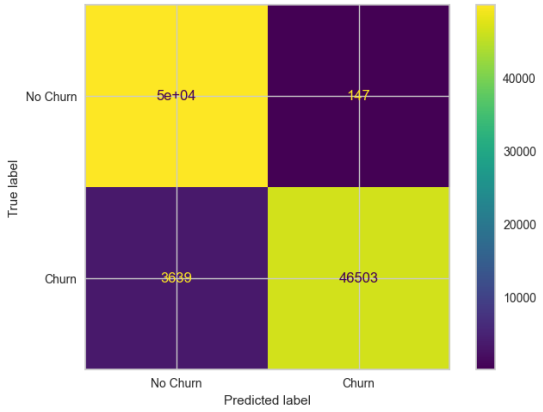

Figure 5.6. Dataset1 Confusion Matrix: KNN

5.4. Discussion

5.4.1. Diabetes Prediction

- Hypothesis 1 was supported as feature selection significantly improved model accuracy.
- Hypothesis 2 was confirmed; using Robust Scaler effectively mitigated the impact of outliers.
- Hypothesis 3 showed improvement in SVM and KNN performance with undersampling. We saw the f1-score moved from 0.6 to 0.85.
- Hypothesis 4 was partially supported, as ANN had competitive performance values but does not have the higher score than SVM.

5.4.2. Churn Prediction

- Hypothesis 1 was validated; oversampling improved performance across models.
- Hypothesis 2 was confirmed, with RBF kernel outperforming other kernels in SVM.
- Hypothesis 3 was supported; KNN showed superior performance due to clear feature separability.

6. Conclusion

This comparative analysis highlights the performance nuances of Neural Networks, Support Vector Machines, and k-Nearest Neighbors across two distinct datasets. Overall, churn dataset was easily learnable across all machine learning models. This was mainly due to clear class separation, lesser features (dimensions), and the data pre-processing done before fitting the model such as oversampling and robust scaling. The initial results for Diabetes prediction showed bad scores for SVM and KNN, while ANN without any data processing was still able to give a F1-score of 0.75. But after we applied undersampling and the right hyperparameters, we saw improvement across all models.

SVM marginally outperformed the other model for dataset1, whereas it underperformed for Dataset2. This shows its robustness in handling imbalanced data with outliers, but for simpler data, it cannot outperform KNN.

ANN also performed well, benefiting from fine-tuning of learning rate and activation functions.

KNN provided competitive performance, particularly in scenarios with clear separations.

Overall, while each algorithm has its strengths, the choice of model should be guided by the specific characteristics of the dataset and the importance of different evaluation metrics. Future work may involve deeper analysis of hyperparameter tuning and exploring additional datasets for a more comprehensive understanding.

References

[1] H. Cover T. M., *Nearest neighbor pattern classification. ieee transactions on information theory*, 1967.

[2] V. Cortes Vapnik, *Support-vector networks. machine learning, 20(3), 273-297*, 1995.

[3] C. Goodfellow Bengio, *Deep learning. mit press.* 2016.

[4] *Cdc diabetes health indicators*. [Online]. Available: https://archive.ics.uci.edu/dataset/891/cdc+diabetes+health+indicators.

[5] *Customer churn dataset*. [Online]. Available: https://www.kaggle.com/datasets/muhammadshahidazeem/customer-churn-dataset?select=customer_churn_dataset-testing-master.csv.

[6] B. B. C. Shiv Ram Dubey1 Satish Kumar Singh1, *Activation functions in deep learning: A comprehensive survey and benchmark*. [Online]. Available: https://ar5iv.labs.arxiv.org/html/2109.14545.