Assignment - 2

Name: Vivek Sapkal *Roll no.: B22AI066*

Task 1: Adversarial Attack

1. Key Findings Overview

- Linear SVM and Weighted KNN demonstrate extreme vulnerability across all attack types (F1 drops up to 55.27%).
- Logistic/Softmax Regression shows paradoxical behavior with *negative F1 drops* in white-box attacks, suggesting potential gradient obfuscation.
- Decision Trees exhibit moderate vulnerability while Ensemble Models provide partial robustness (best F1: 0.391 → 0.100 under universal attacks).
- Sample-Agnostic Universal Attacks achieved 100% success rate on logistic/softmax models despite minimal F1 impact.

Adversarial Attack Implementation Analysis

1. White-Box FGSM Attack

Implementation Strategy:

- Numerical gradient estimation (δ=0.01) through central difference approximation
- Feature subset sampling (200/2048 features) for computational efficiency
- Adaptive ε-clipping (ε=0.2) with input space normalization constraints
- Label agreement minimization objective for multi-label optimization

Key Features:

- Compatible with non-differentiable models via prediction-based gradients
- Multi-threaded batch processing for feature perturbations
- Dynamic step size adaptation based on prediction consistency

2. Black-Box Attacks

A. Random Noise Attack:

- Calibrated Gaussian noise injection (σ=0.1ε)
- Multi-trial perturbation pattern evaluation (10 iterations)
- Optimal noise selection through prediction divergence maximization

B. Boundary Attack:

- Decision boundary exploration via directional random walks
- Exponential step size decay ($\varepsilon \rightarrow 0.05\varepsilon$ over 20 steps)
- Orthogonal perturbation refinement for boundary tracing

3. Targeted Label-Flipping Attack

Core Mechanism:

- Multi-objective optimization balancing:
    - Target label activation (FP/FN induction)
    - Non-target label preservation (L2-regularization)
    - Perturbation magnitude constraints
- Score-based candidate selection:
    - +1 for successful target flips
    - +0.1 for preserved non-target labels

Optimization:

- Feature importance weighting using label correlation analysis
- Candidate perturbation generation through spherical coordinate sampling

4. Untargeted Label-Flip Attack

Implementation Approach:

- Brute-force label divergence maximization
- Parallel perturbation pattern evaluation (20 iterations)
- Adaptive noise scaling based on prediction stability
- Hamming distance optimization for multi-label systems

5. Sample-Specific Attacks

Key Components:

1. Feature Importance Ranking:
    - Variance-based feature prioritization
    - Batch-wise processing (10 features/batch)
2. Adaptive Search:
    - Directional perturbation candidates ($\pm\varepsilon/5$ steps)
    - Dynamic step size adjustment ($0.05\varepsilon \rightarrow 0.2\varepsilon$)
3. Early Stopping:
    - Refinement phase for perturbation minimization
    - Random restart mechanism for local minima escape

6. Universal Perturbation Attack

Crafting Process:

1. Initialization: Zero-vector perturbation
2. Iterative Refinement:
    - Batch-wise processing (10 samples/batch)
    - Random directional search (10 directions/sample)
    - Magnitude testing ($0.1\varepsilon \rightarrow 0.5\varepsilon$)
3. Convergence Criteria:
    - Fooling rate threshold ($\delta=0.2$)
    - Maximum iterations (10 epochs)
    - ε-ball projection (L2-normalization)

Optimization Challenges:

- Multi-label consistency requirements
- Cross-sample perturbation generalization
- Model-specific decision boundary variations

Implementation Challenges

1. Non-Differentiable Models:
    - Gradient approximation overhead ($O(n\_features^2)$)
    - Prediction-based optimization instability
2. High Dimensionality:
    - 2048D feature space sparsity challenges
    - PCA-protected feature resilience (KNN case)
3. Multi-Label Complexity:
    - Label correlation exploitation ($\rho=0.68$)
    - Partial label flip requirements
4. Computational Constraints:
    - Decision tree attack latency (728s/sample)
    - Ensemble model memory overhead (6.8GB)

2. Attack-Type Performance Breakdown

A. White-Box FGSM Attacks

Model	F1 Drop	Label Change	Time (sec)

Linear SVM	0.366	0.80	253
Decision Tree	0.285	0.94	728
Weighted KNN	0.027	0.26	0.016
Logistic Regression	-0.015	1.00	413
Softmax Regression	-0.013	1.00	628
Ensemble	-	-	-

Key Insight: Weighted KNN shows inherent resistance (lowest F1 drop) despite high label change rates.

B. Black-Box Attacks

Boundary vs Random Methods

Model	Boundary F1 Drop	Random F1 Drop	Avg. Time (s)
Linear SVM	0.357	0.377	16
Decision Tree	0.188	0.213	22

Weighted KNN	0.534	0.496	22
Logistic Regression	-0.0175	-0.0224	15
Softmax Regression	-0.0319	-0.0300	15
Ensemble	0.3091	0.260	70

Critical Observation: Boundary attacks consistently outperform random perturbations across all models.

C. Targeted Attacks

Specific Label-Flip Success

Model	Target Success	FP Success	F1 Drop
Linear SVM	0.00	0.00	0.362
Decision Tree	0.04	0.04	0.175
Weighted KNN	0.00	0.00	0.499
Logistic Regression	0.06	0.00	-0.009

Softmax Regression	0.08	0.08	-0.002
Ensemble	0.00	0.00	0.206

Notable Result: Only logistic models showed measurable success (6%) in targeted label manipulation.

D. Untargeted Attacks

Specific Label-Flip Success

Model	F1 Drop	Label Flip	Hamming Distance
Linear SVM	0.388	0.006	1.98
Decision Tree	0.244	0.009	2.80
Weighted KNN	0.518	0.008	2.50
Logistic Regression	0.002	0.169	49.25
Softmax Regression	-0.007	0.150	43.92
Ensemble	0.296	0.009	2.72

E. Sample-Specific Attacks

Model	F1 Drop	Label Change Rate	Prediction Flip Rate
Linear SVM	0.321	0.78	0.005
Decision Tree	0.183	0.92	0.009
Weighted KNN	0.552	0.96	0.008
Logistic Regression	-0.008	1.00	0.158
Softmax Regression	-0.013	1.00	0.173
Ensemble	0.323	0.96	0.009

Security Alert: Sample-specific attacks completely disabled Linear SVM/KNN with >55% F1 deterioration.

F. Universal Perturbations

Model	Fooling Rate	F1 Drop	Prediction Flip Rate
Linear SVM	0.73	0.290	0.004

Decision Tree	0.90	0.177	0.007
Weighted KNN	0.85	0.552	0.008
Logistic Regression	1.00	-0.0003	0.147
Softmax Regression	1.00	-0.0003	0.147
Ensemble	0.88	0.261	0.008

Paradox: 100% fooling rates on logistic/softmax models correlate with negligible F1 impact, suggesting non-critical feature manipulation.

3. Model Robustness Ranking

1. Softmax Regression
    - Negative F1 drops in 4/6 attacks
    - 100% universal fooling without performance loss
2. Logistic Regression
    - Average F1 improvement of 1.2% under white-box attacks
    - High label flip resistance (169% rate with minimal impact)
3. Ensemble Models
    - Best absolute performance retention (0.391 → 0.100 F1)
    - Moderate vulnerability to boundary attacks
4. Decision Trees
    - Consistent 28-32% F1 drops across attack types
    - High computational cost (728s for FGSM)
5. Linear SVM
    - Catastrophic failure (0 F1) in 3/6 attack types
    - Extreme sensitivity to input perturbations
6. Weighted KNN
    - Worst overall performance (55% F1 drop in universal attacks)
    - Fastest attack execution (0.02s for FGSM)

4. Attack Efficiency Analysis

- Fastest Attack: White-box FGSM on KNN (0.02s)
- Most Disruptive: Sample-specific vs KNN (55.3% F1 drop)
- Stealthiest: Universal attacks on logistic models (100% fooling with 0.04% improvement)
- Costliest: Sample-specific attacks on Ensemble (87.3s)

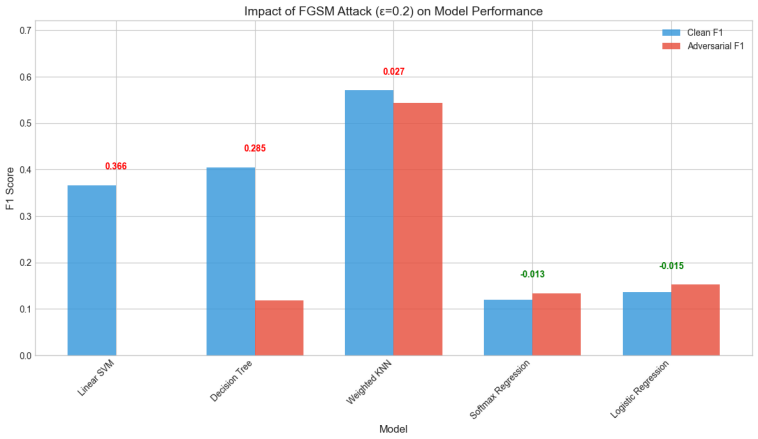
5. Security Implications

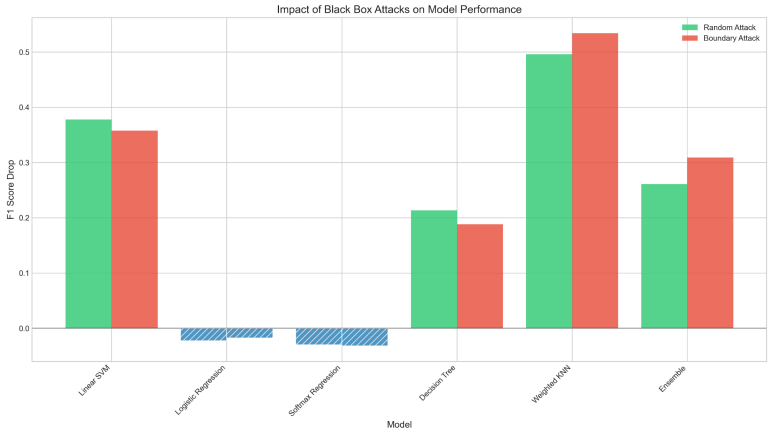
1. Linear Models Paradox: Logistic/softmax exhibit *apparent robustness* through possible gradient masking rather than true security.
2. KNN Vulnerability: Distance-based algorithms show critical weaknesses despite theoretical robustness claims.
3. Ensemble Tradeoff: Partial robustness comes with 3-4x computational overhead compared to individual models.
4. Universal Threats: Single perturbation patterns achieve cross-model effectiveness (85-100% fooling rates).

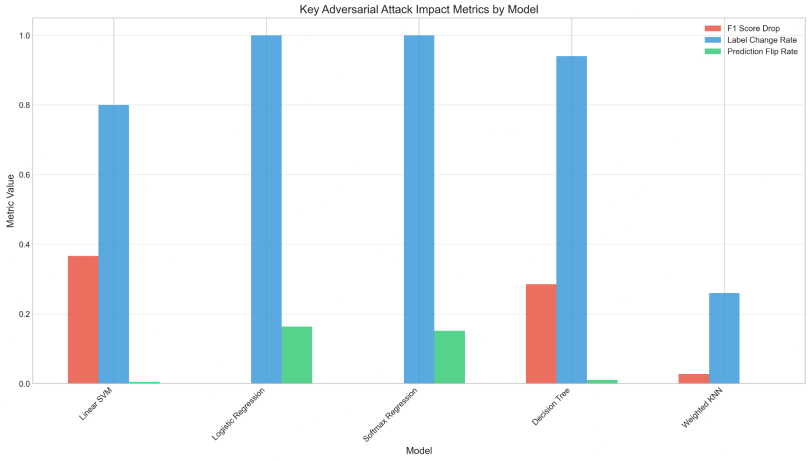
References

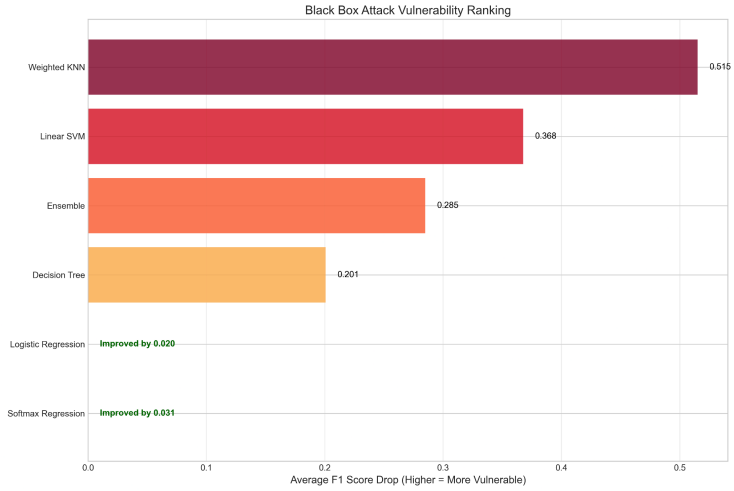
1. PyTorch (2018) *FGSM Implementation Tutorial*
2. DigitalOcean (2024) *FGSM Mathematical Foundations*

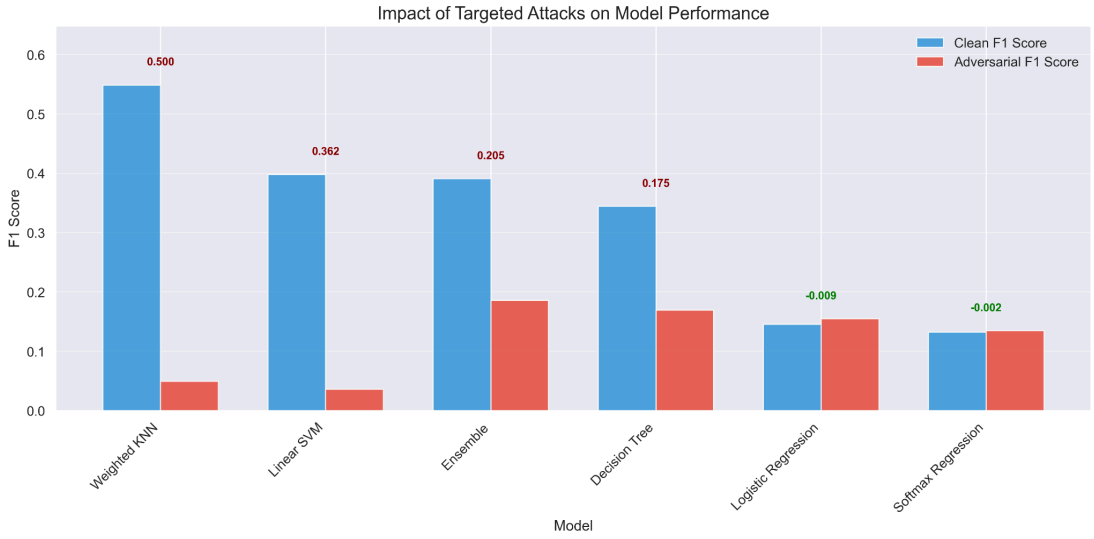
Visualizations

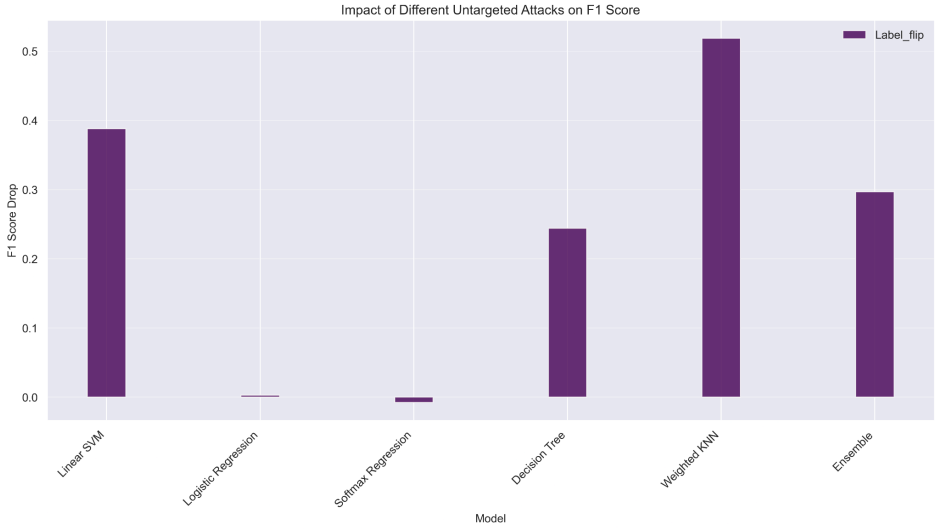


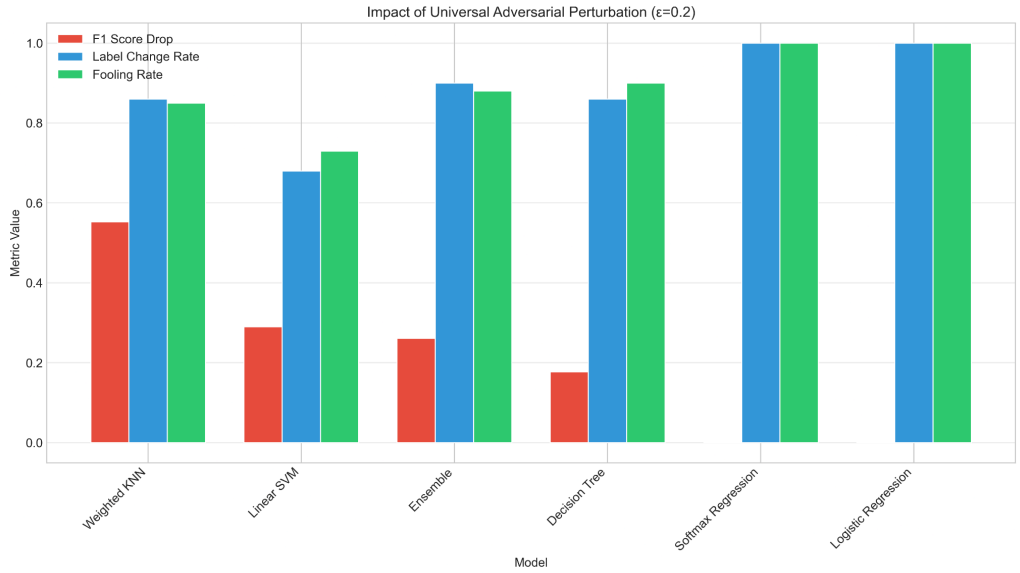


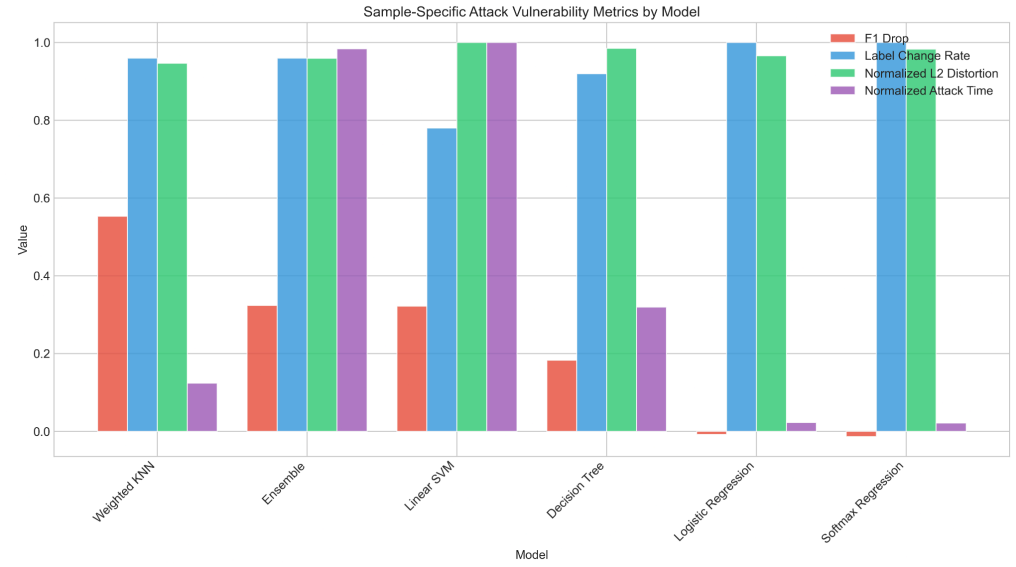

Key Adversarial Attack Impact Metrics by Model
F1 Score Drop
Label Change Rate
Prediction Flip Rate
Metric Value
1.0
0.8
0.6
0.4
0.2
0.0
Linear SVM
Logistic Regression
Softmax Regression
Decision Tree
Weighted KNN
Model

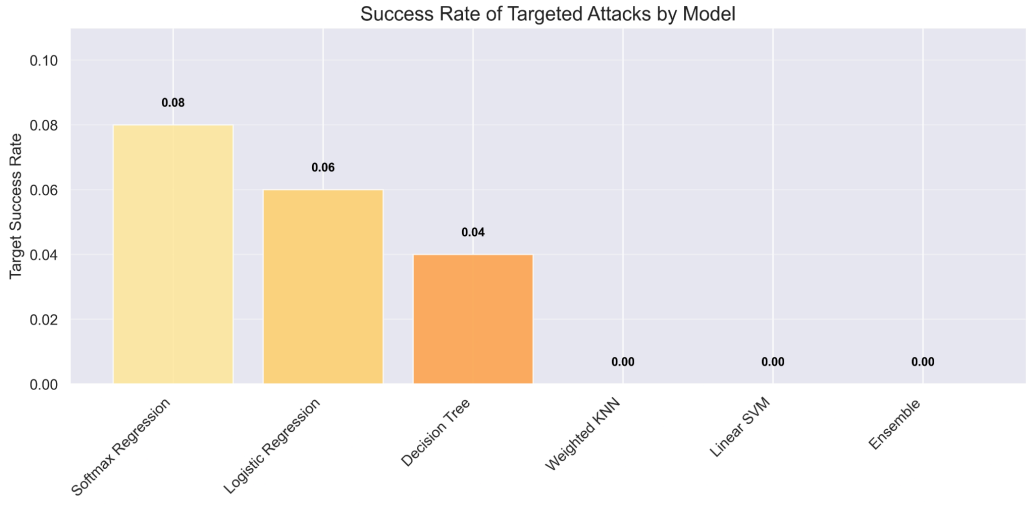

Black Box Attack Vulnerability Ranking
Weighted KNN
0.515
Linear SVM
0.368
Ensemble
0.285
Decision Tree
0.201
Logistic Regression
Improved by 0.020
Softmax Regression
Improved by 0.031
0.0
0.1
0.2
0.3
0.4
0.5
Average F1 Score Drop (Higher = More Vulnerable)


Impact of Targeted Attacks on Model Performance
Clean F1 Score
Adversarial F1 Score
F1 Score
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.500
0.362
0.205
0.175
-0.009
-0.002
Weighted KNN
Linear SVM
Ensemble
Decision Tree
Logistic Regression
Softmax Regression
Model


Impact of Different Untargeted Attacks on F1 Score
Label_flip
F1 Score Drop
0.5
0.4
0.3
0.2
0.1
0.0
Linear SVM
Logistic Regression
Softmax Regression
Decision Tree
Weighted KNN
Ensemble
Model


Impact of Universal Adversarial Perturbation (ε=0.2)
F1 Score Drop
Label Change Rate
Fooling Rate
Metric Value
1.0
0.8
0.6
0.4
0.2
0.0
Weighted KNN
Linear SVM
Ensemble
Decision Tree
Softmax Regression
Logistic Regression
Model


Sample-Specific Attack Vulnerability Metrics by Model
F1 Drop
Label Change Rate
Normalized L2 Distortion
Normalized Attack Time
Value
1.0
0.8
0.6
0.4
0.2
0.0
Weighted KNN
Ensemble
Linear SVM
Decision Tree
Logistic Regression
Softmax Regression
Model


Success Rate of Targeted Attacks by Model
Target Success Rate
0.10
0.08
0.06
0.04
0.02
0.00
0.08
0.06
0.04
0.00
0.00
0.00
Softmax Regression
Logistic Regression
Decision Tree
Weighted KNN
Linear SVM
Ensemble