Predicting Stock Movement with Neural Networks

—— A Data-Driven Approach

Using AAPL Return Trends

Nanzhu Li
Stephanie Daniella Hernandez Prado
Serena Zhou

Why Predict Stock Trends?

Traditional Financial Theories: Efficient Market Hypothesis or Random Walk

Behavioral finance

1. **Momentum Effect:** Past winners tend to keep winning
2. Confirmation Bias: Investors focus on info that supports their beliefs
3. Herding Behavior: People follow the crowd, not fundamentals

So why are we doing this project? → Markets are not always rational!

Our goal :

We use historical Apple Inc. (AAPL) data to develop a model that classifies short-term stock price movements

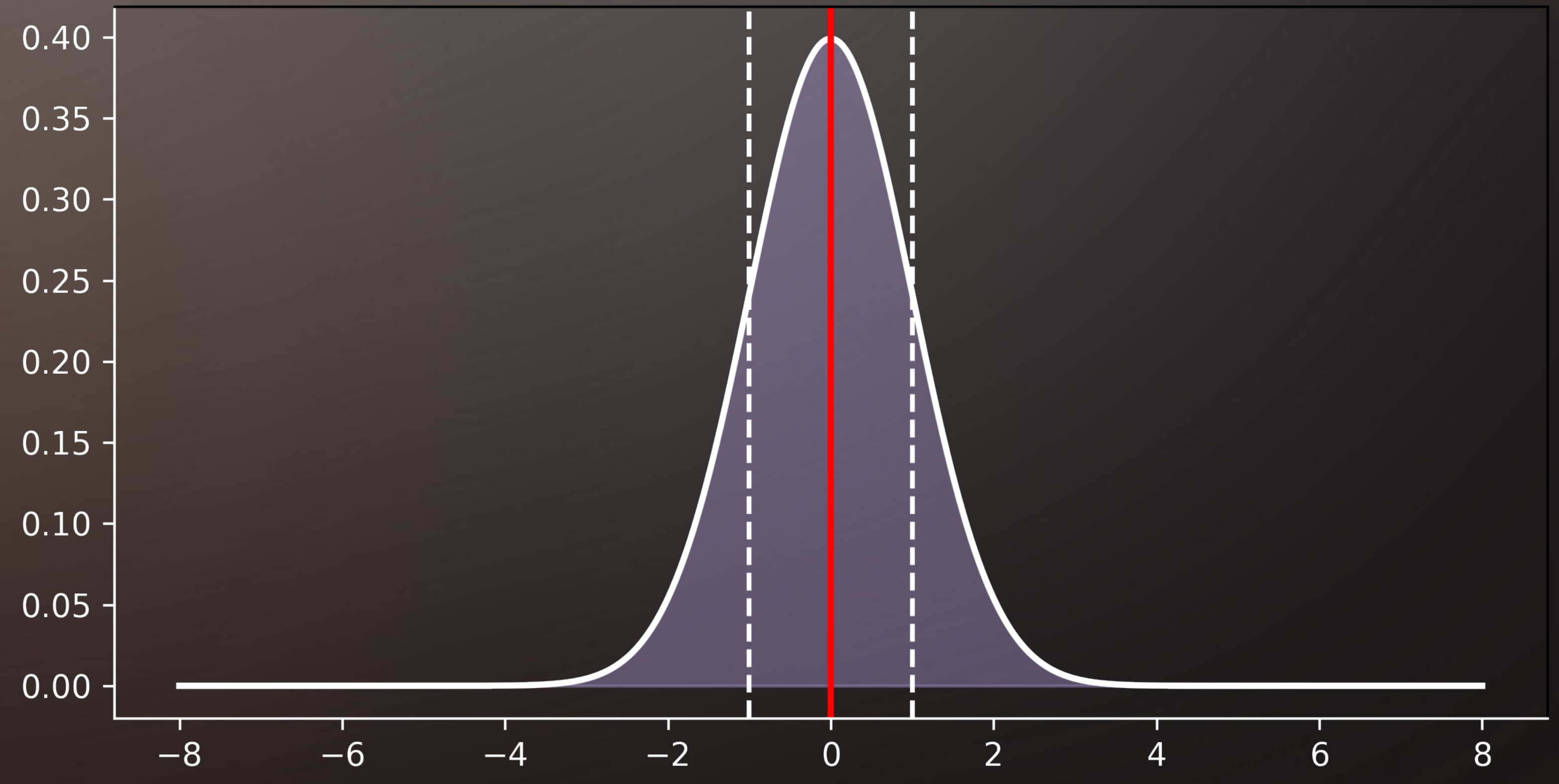
✓ historical Apple Inc. (AAPL) data collected from 2012 to 2025 collected via the yfinance Python library

Assumptions

1. **Future trends depend on past 15-day return patterns.**

   Daily return = $(P_t - P_{i-1}) / P_{i-1}$

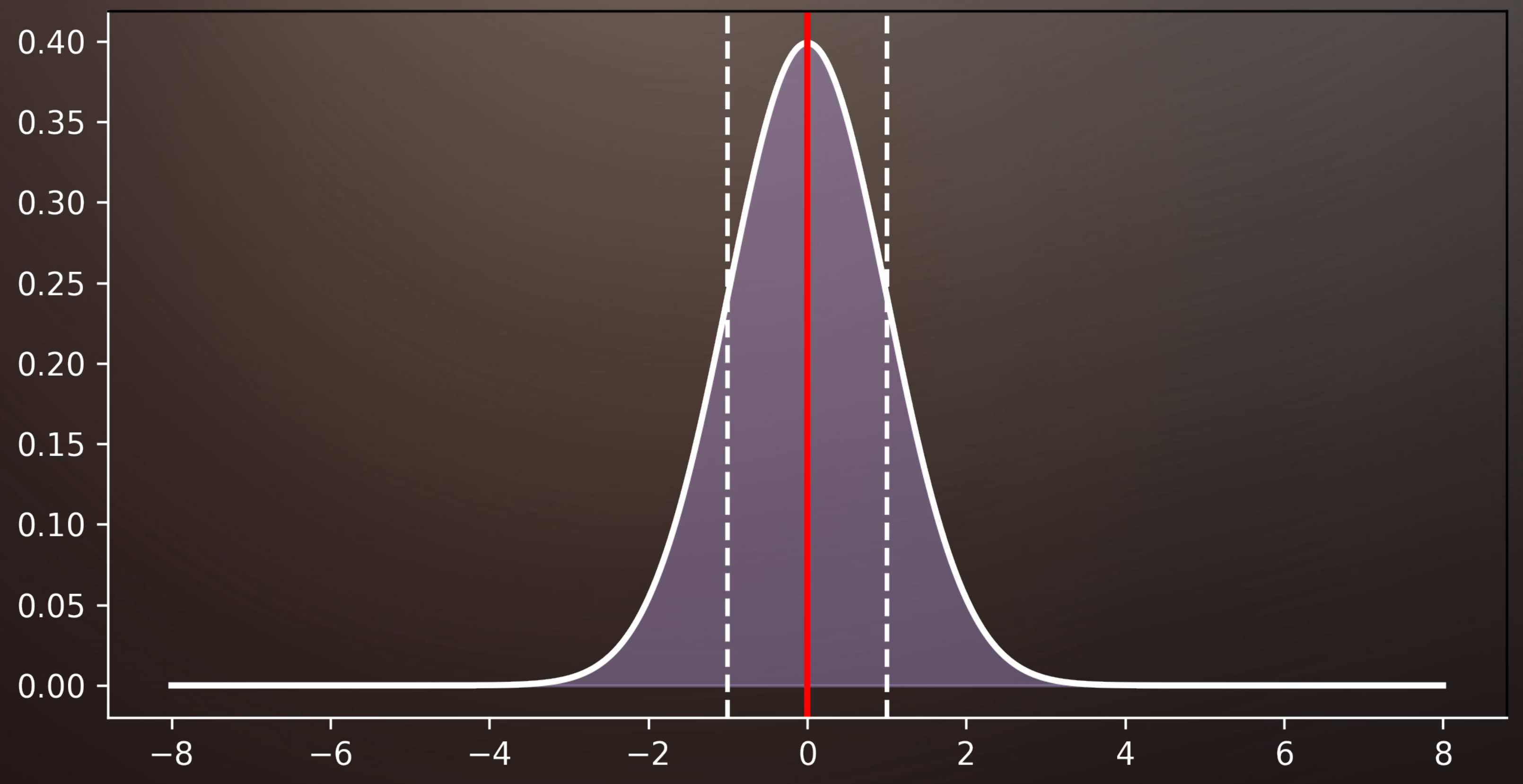
2. **Stock daily returns are assumed to follow a Normal Distribution.**



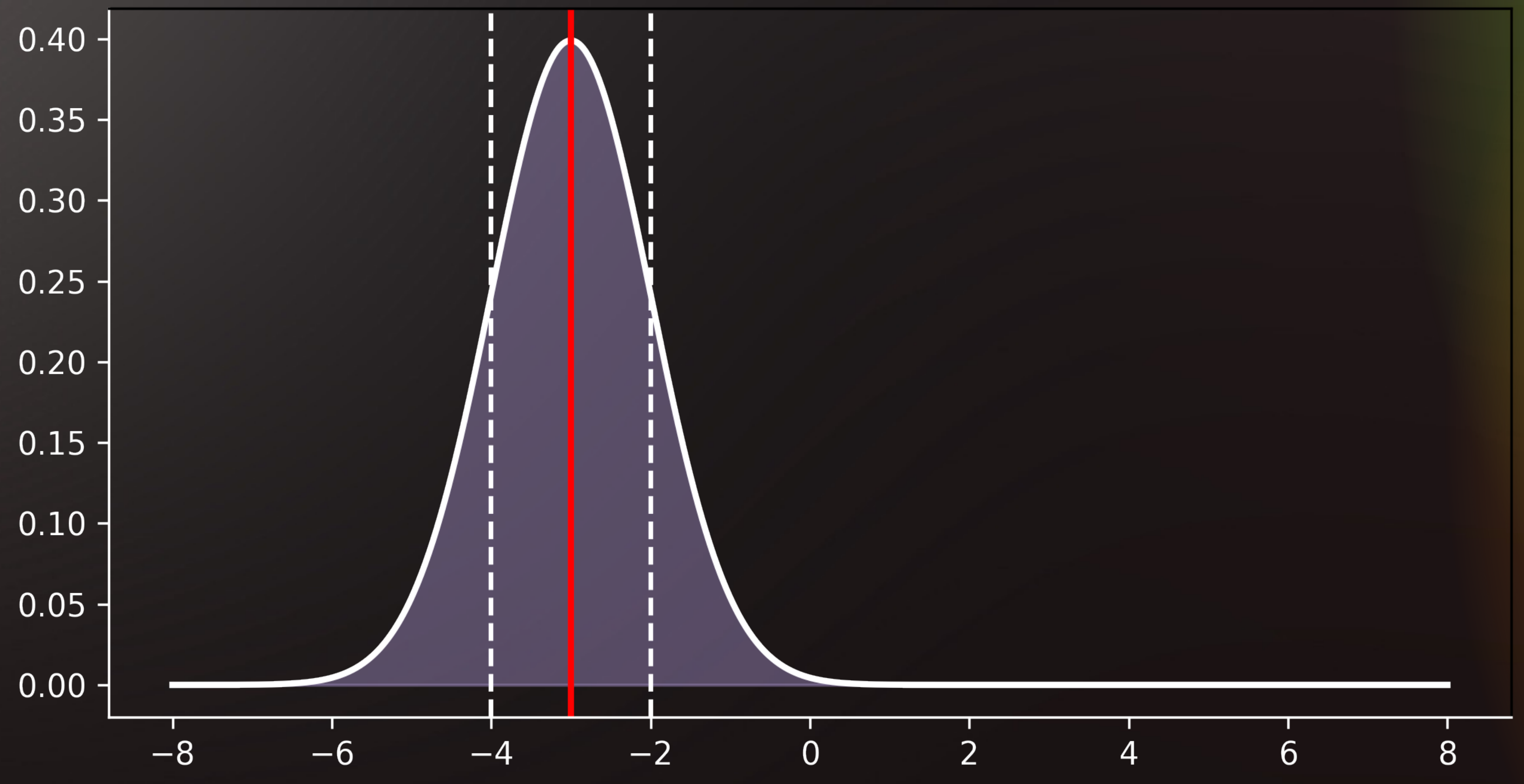
Distribution centered at 0

Labeling the Trend

1. **Look ahead 5 days after each window**
2. **Using the sum of next 5 days to determine label:**
   - ➢ If the rolling sum > 0.3 × std Uptrend 2
   - ➢ If the rolling sum < -0.3 × std Downtrend 0
   - ➢ Otherwise, Neutral 1



Distribution of daily return centered at 0



Distribution centered at -3

Labeling Market Movements

0: “red” Downtrend   1: “yellow” Neutral   2: “green” Uptrend

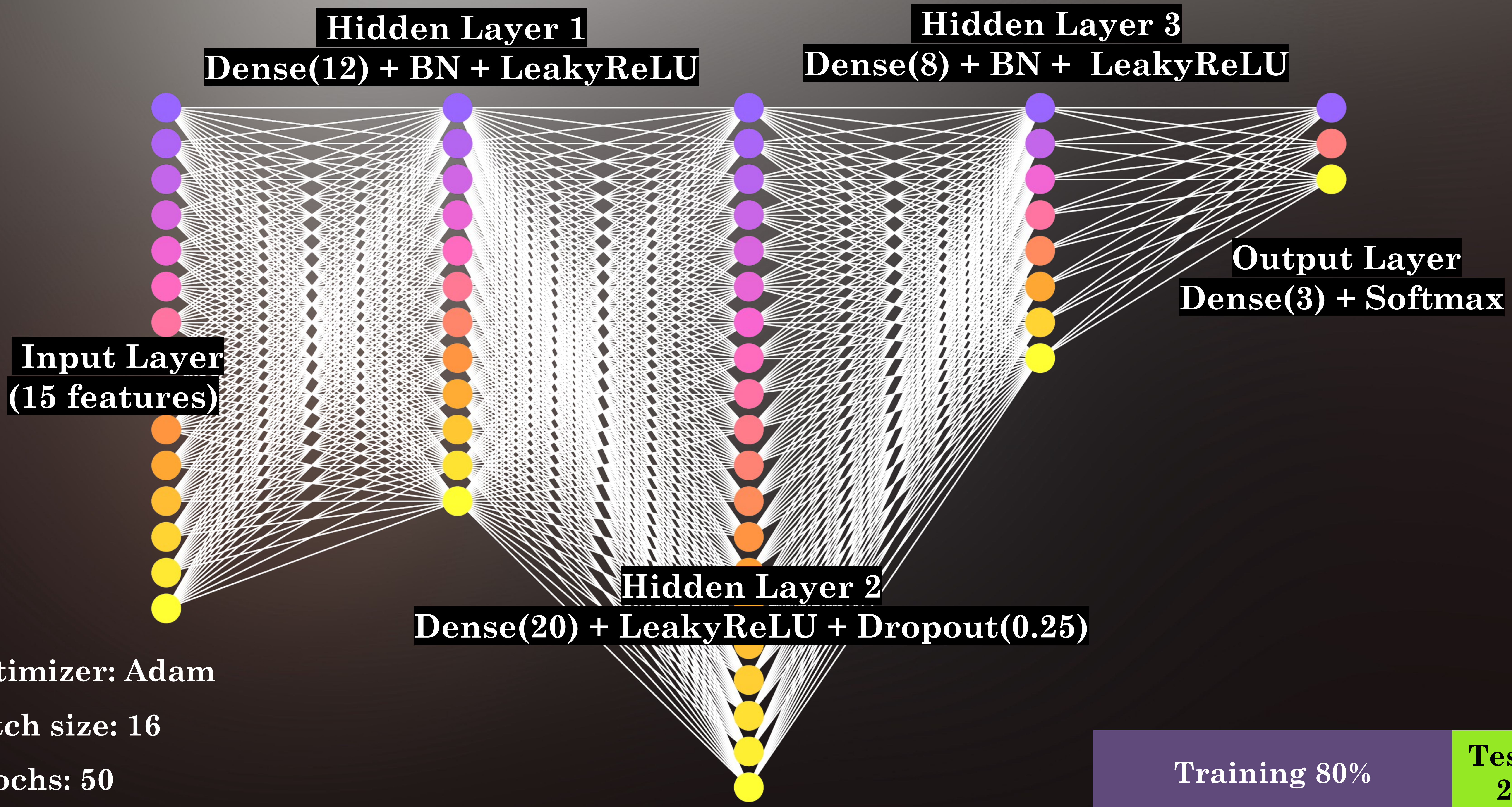
Dataset

Sliding window : 15 consecutive daily returns

Change_1	Change_2	Change_3	Change_4	Change_5	Change_6	Change_7	Change_8	Change_9	Change_10	Change_11	Change_12	Change_13	Change_14	Change_15	Trend
0.005374	0.011102	0.010454	-0.001586	0.003581	-0.001630	-0.002745	-0.003749	0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	2.0
0.011102	0.010454	-0.001586	0.003581	-0.001630	-0.002745	-0.003749	0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	2.0
0.010454	-0.001586	0.003581	-0.001630	-0.002745	-0.003749	0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	1.0
-0.001586	0.003581	-0.001630	-0.002745	-0.003749	0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	0.012811	2.0
0.003581	-0.001630	-0.002745	-0.003749	0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	0.012811	0.007660	1.0
-0.001630	-0.002745	-0.003749	0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	0.012811	0.007660	-0.000635	2.0
-0.002745	-0.003749	0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	0.012811	0.007660	-0.000635	-0.002345	2.0
-0.003749	0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	0.012811	0.007660	-0.000635	-0.002345	0.010019	2.0
0.011649	0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	0.012811	0.007660	-0.000635	-0.002345	0.010019	0.009332	2.0
0.010383	-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	0.012811	0.007660	-0.000635	-0.002345	0.010019	0.009332	0.010475	2.0
-0.003169	-0.017417	0.016917	-0.016378	0.062439	-0.004545	0.005960	0.012811	0.007660	-0.000635	-0.002345	0.010019	0.009332	0.010475	0.016744	2.0

This process generated 3,304 labeled samples, each with 15 input features.

Neural Networks



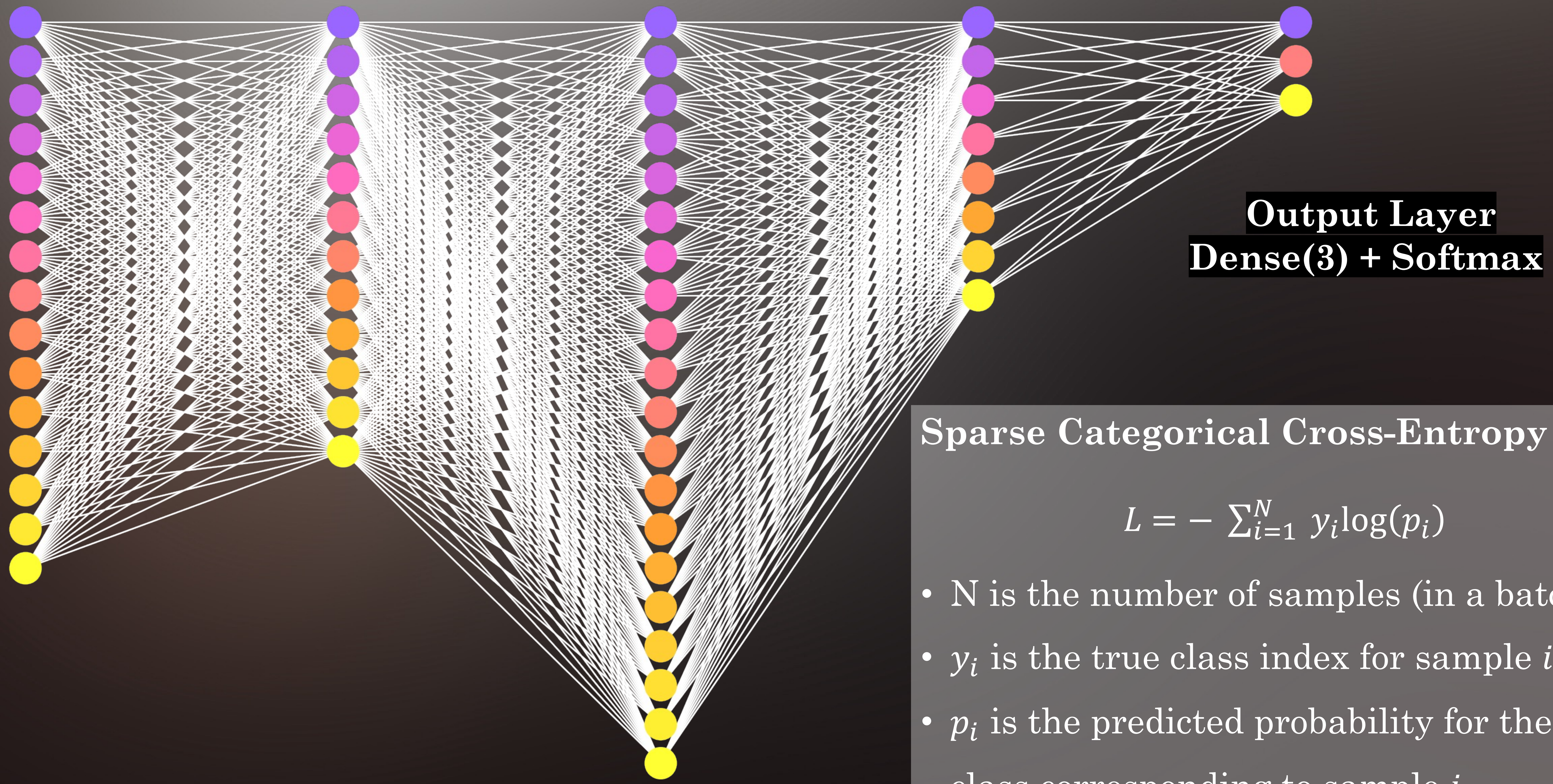
Optimizer: Adam

Batch size: 16

Epochs: 50

Training 80%	Testing 20%

Sparse Categorical Cross-Entropy

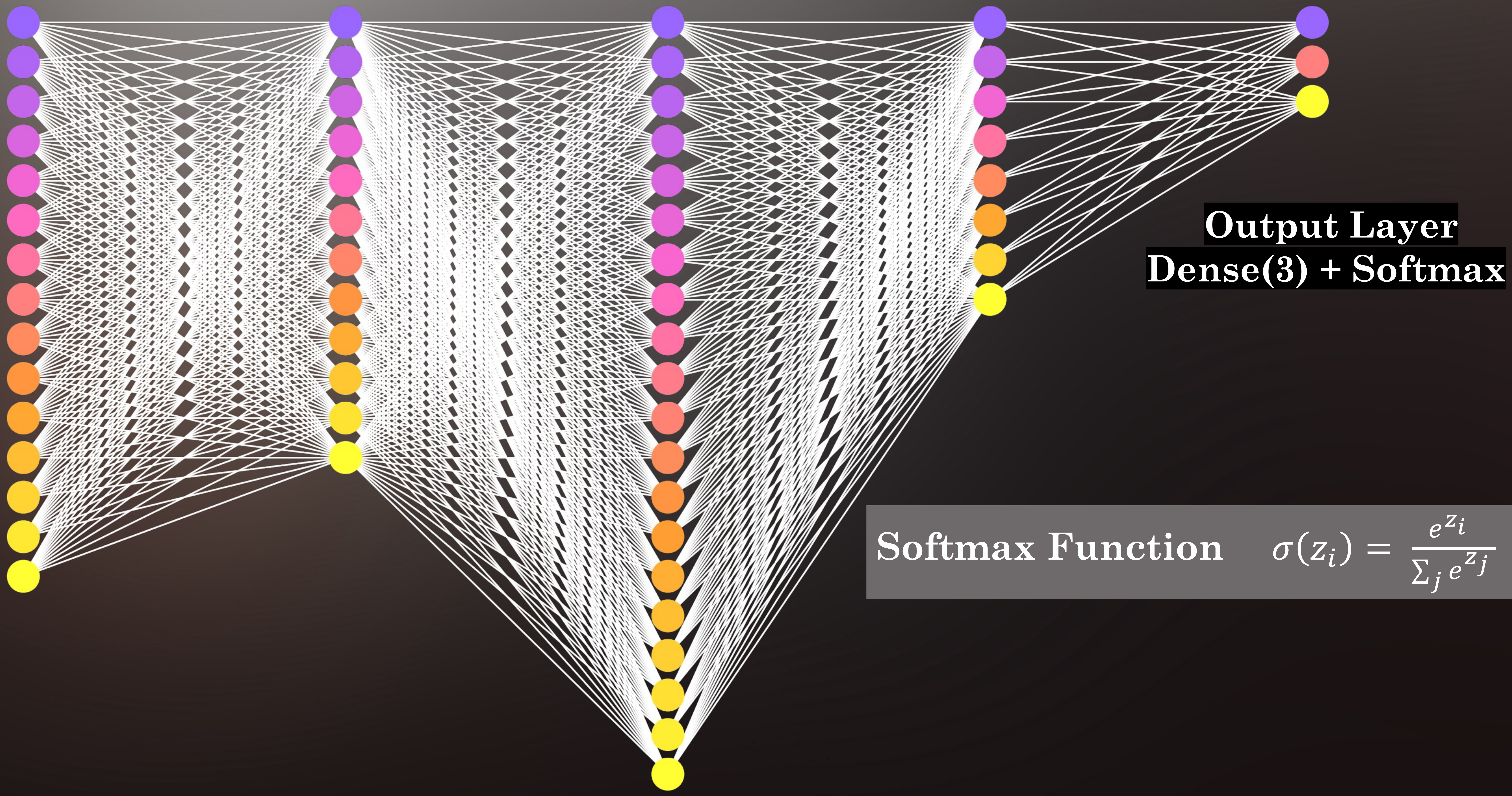


Sparse Categorical Cross-Entropy

$$L = -\sum_{i=1}^{N} y_i \log(p_i)$$

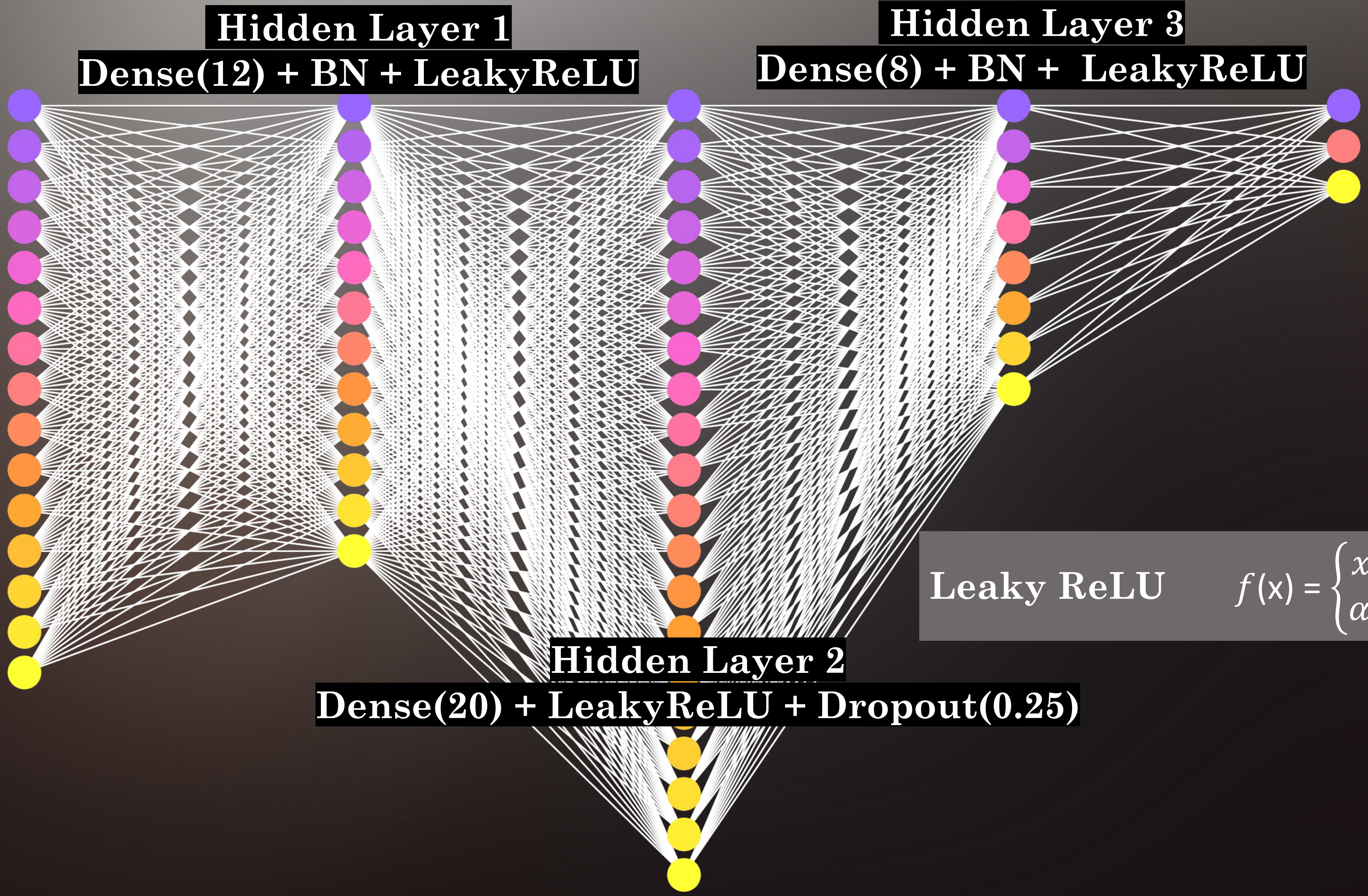
- N is the number of samples (in a batch)
- y_i is the true class index for sample i
- p_i is the predicted probability for the true class corresponding to sample i

Softmax Function



Softmax Function $\sigma(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$

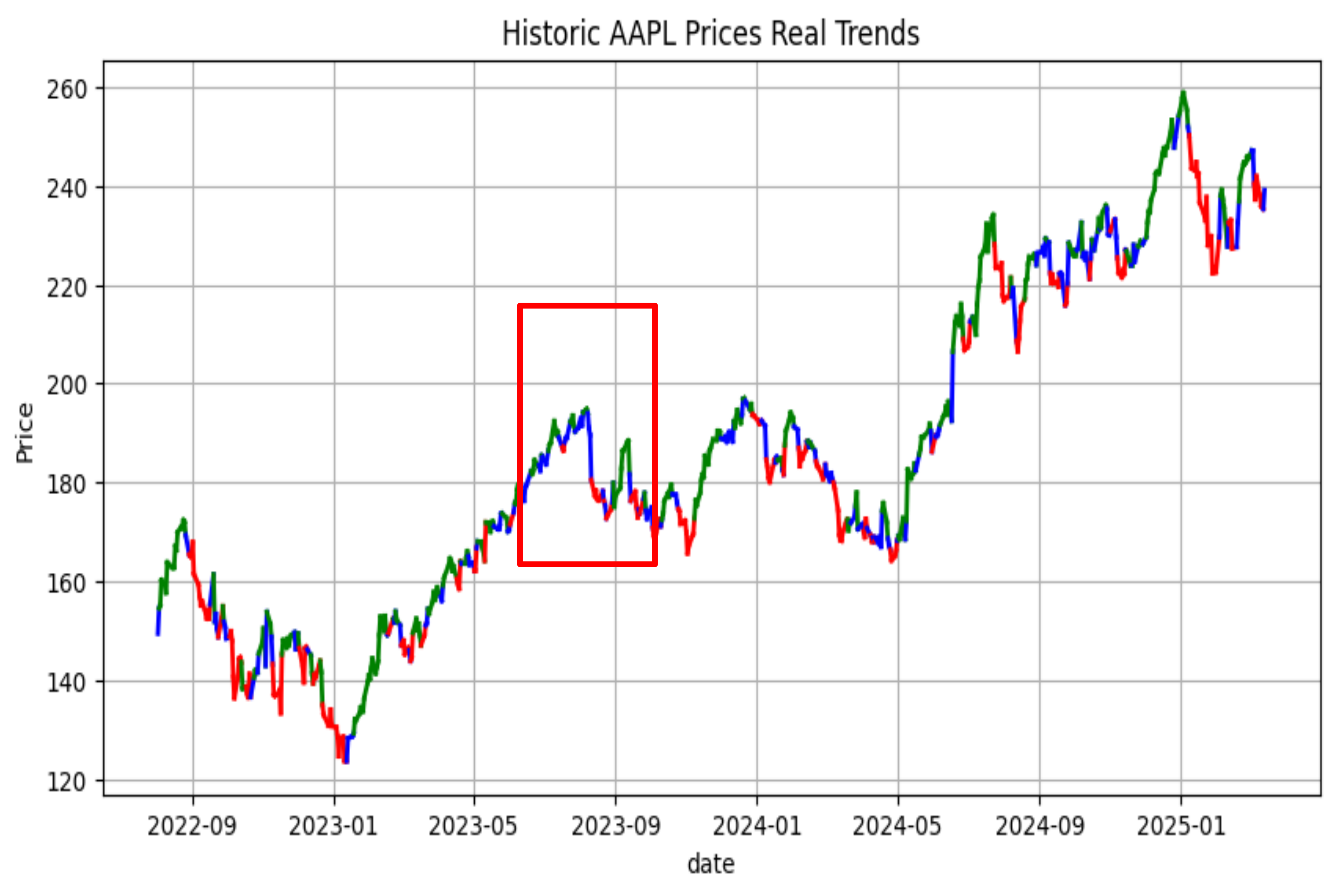
Leaky ReLU

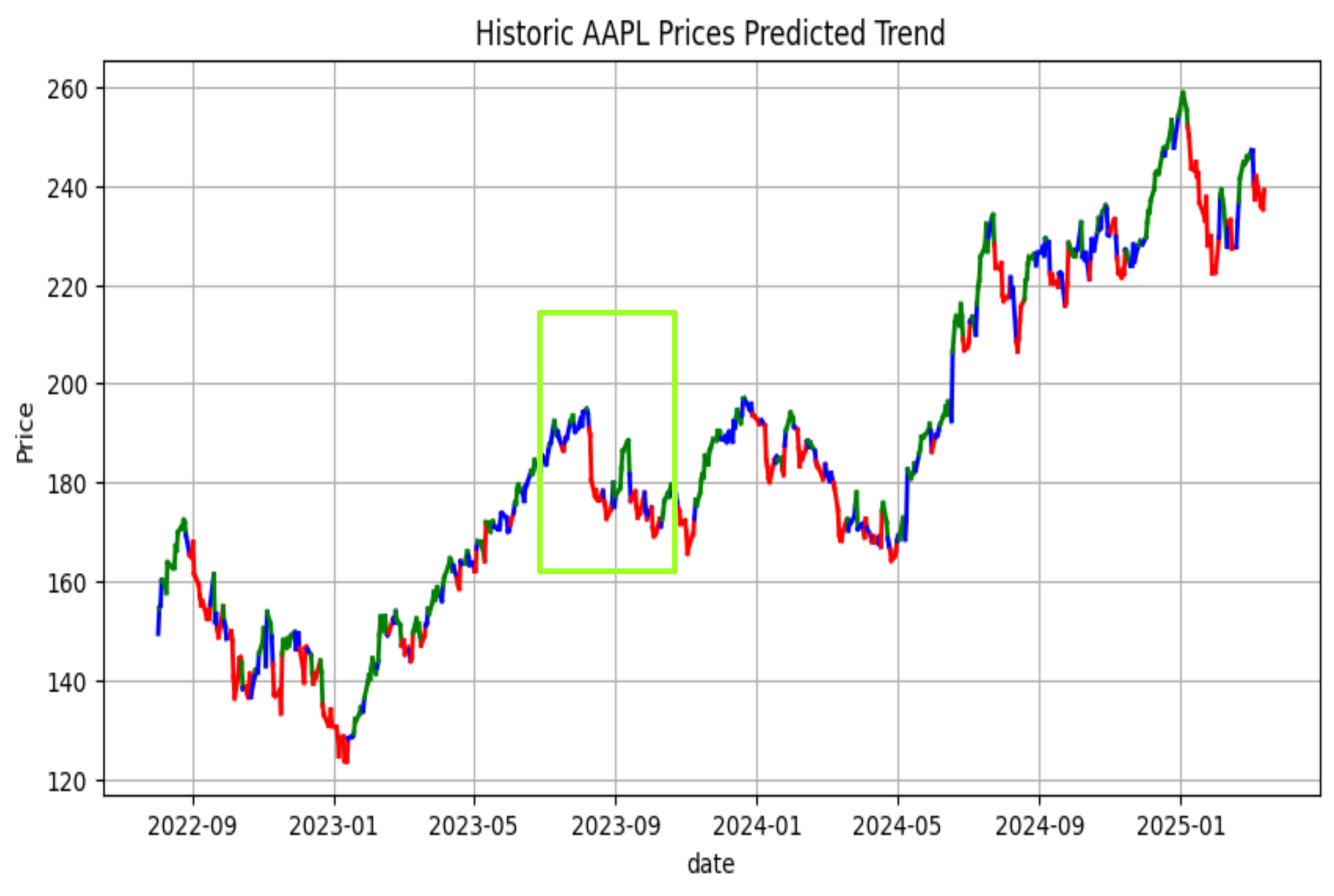


Leaky ReLU $f(x) = \begin{cases} x & if\ x > 0 \\ \alpha x & if\ x \leq 0 \end{cases}$

Conclusions:

Loss Function= 0.247 Accuracy = 91.32%





0: "red" Downtrend   1: "blue" Neutral   2: "green" Uptrend

Limitations and Further Improvements

Limitations

1. Single-asset focus

The model is trained and tested only on AAPL stock data.

2. Labeling accuracy

The classification of stock movements relies on predefined thresholds, which may oversimplify complex market behaviors.

Future work

1. We could apply the method to more stocks or asset classes to evaluate generalizability.
2. Experiment with dynamic or learned labeling strategies to improve classification accuracy..

Thank you !

Our Github