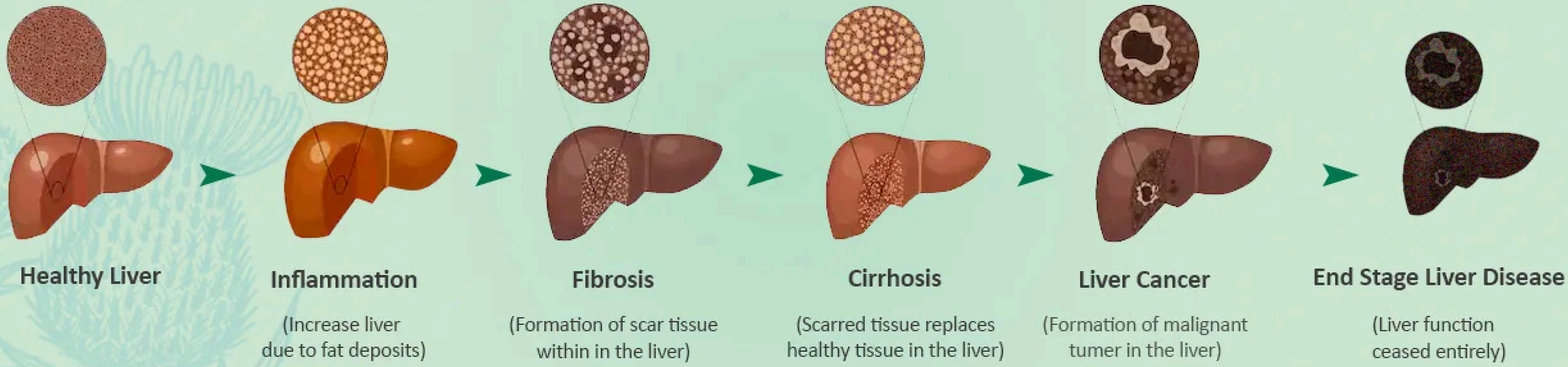
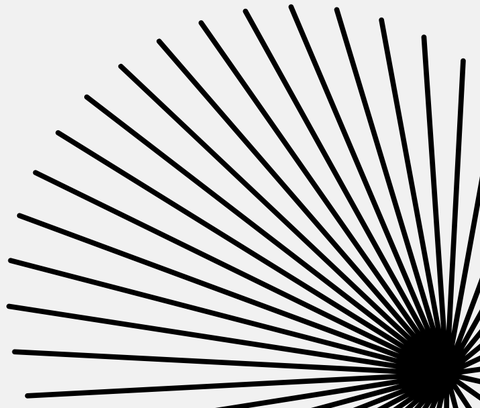



# Survival Analysis of Primary Biliary Cirrhosis

**Jean Baptiste Habyarimana**

# Background

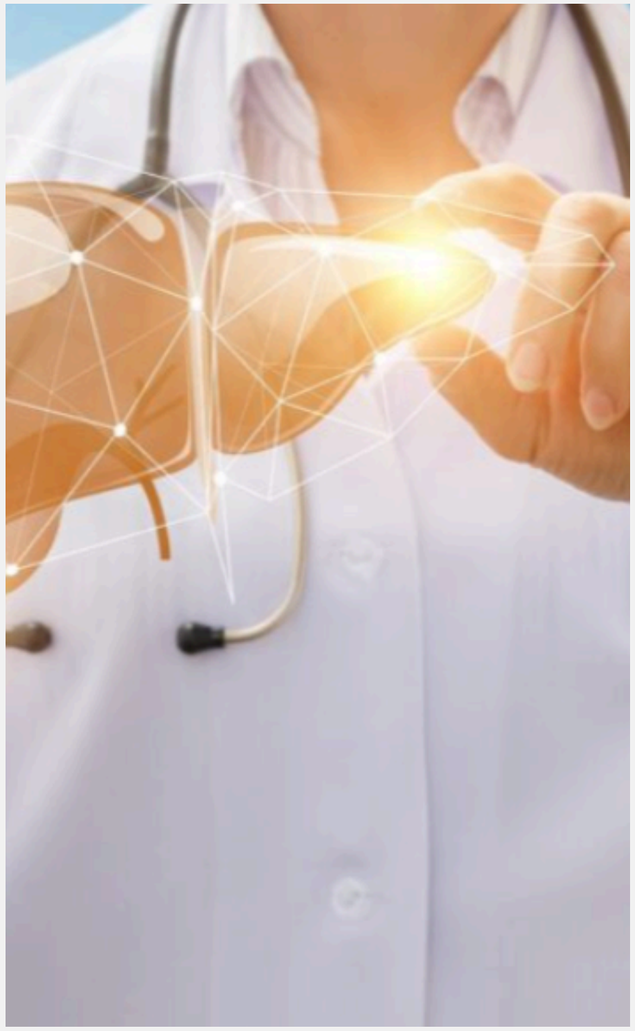
- Dataset from Mayo Clinic conducted between 1974 to 1984 to study Primary Biliary Cirrhosis (PBC) (a chronic liver disease).
- The trial involved 312 patients, and divided into 2 groups.
- One group got the real drug, and the other got a fake pill (placebo).
- Each observation states health status, and liver-related health parameters for each patients
- The goal was testing if a drug called D-penicillamine could help people with PBC.



# Objective and Significance

The goal is to predict patient outcomes (death, liver transplantation, or censored) based on patient characteristics and treatment status. Current work can help:

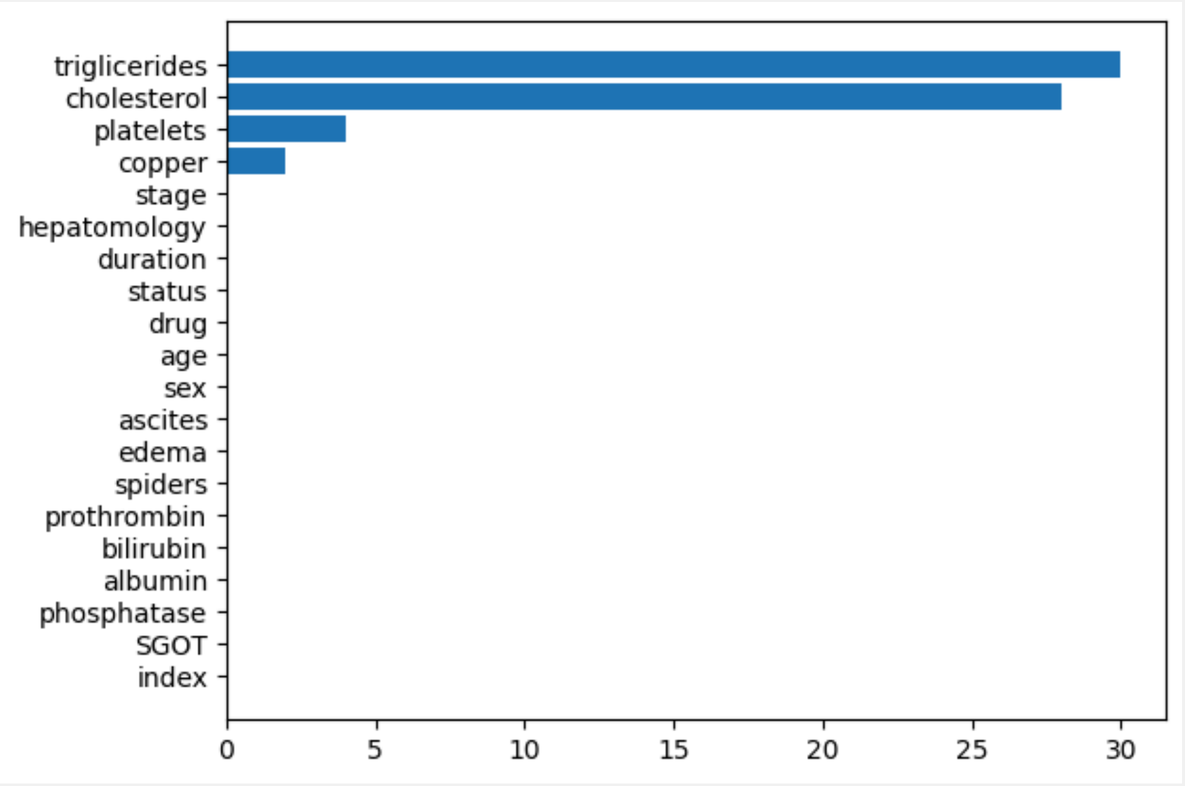
- Predict the patient status and likelihood of death, liver transplant, or survival for each patient using their clinical profile.



- Provide data-driven survival analysis, allowing clinicians to focus on targeted care.
- Allows  hospitals to prioritize resources for high-risk patients, leading to better outcomes and optimized healthcare delivery.
- Insights into drug effectiveness to help pharmaceutical companies and medical professionals make informed decisions about drug approval and use.

# Data Understanding

- Dimension of dataset was (312, 19)
- Mixed data types
- Presence of missing values
- No duplicates




**Target:**

- Survival (0), Death (1), Liver Transplant (2).

**Key Features**

- Ascites: condition where fluid accumulates in the abdomen (0: No, 1: Yes)
- Hepatomology: enlargement of liver (0: No and 1: Yes)
- Spiders: abnormal blood vessels on the skin (0: No and 1: Yes)
- Edema: Fluid accumulation in the legs
- Stage: The histologic stage of the disease, with higher numbers indicating more advanced disease stages.

# Methodology

## Data preprocessing

- Handling missing values.
- Dropping identifiers
- Data transformation (log)
- Features scaling and class balance.
- Outlier handling

## Modeling

- Logistics regression

Multinomial Logistic Regression

- NN

Artificial neural networks

## Result analysis

Accuracy

metric

# Operation parameters

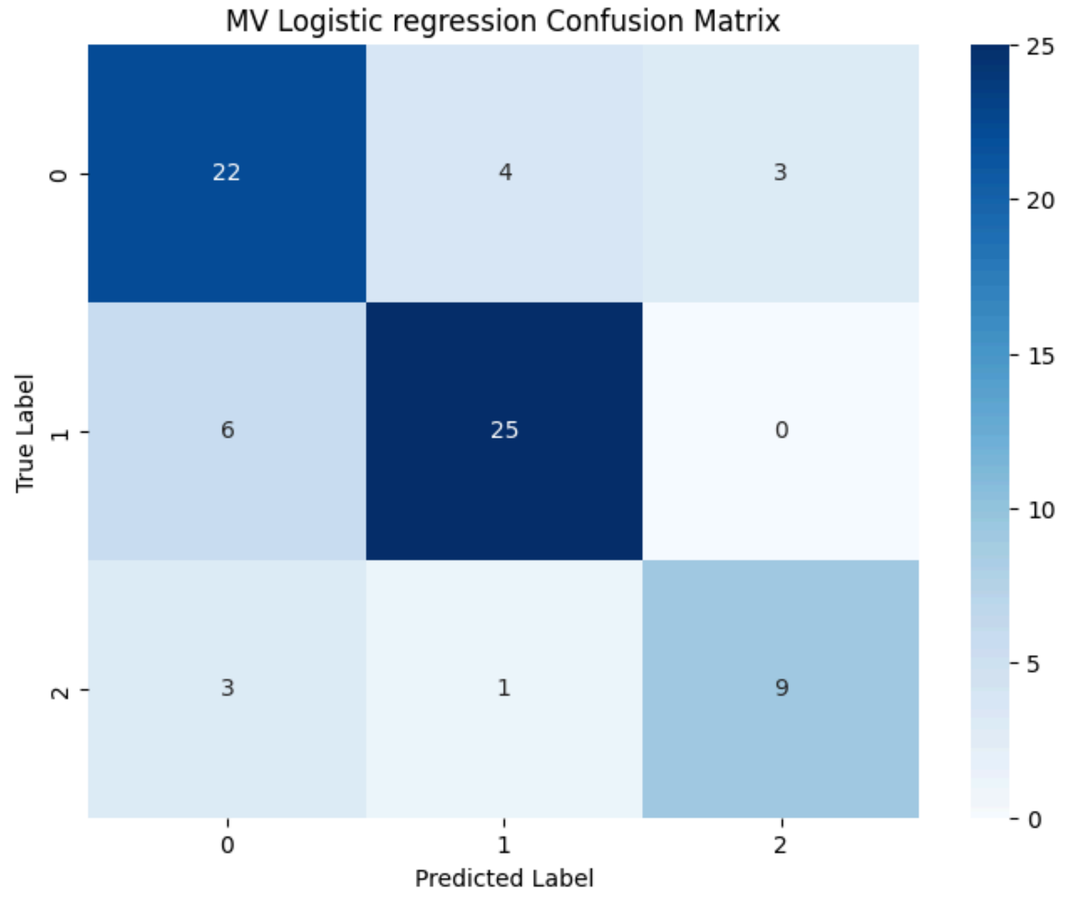
**Logistic regression**
Multi-class logistic regression.
Softmax function to normalize the output.
Cross-entropy as the loss function.
Accuracy and confusion matrix for performance evaluation

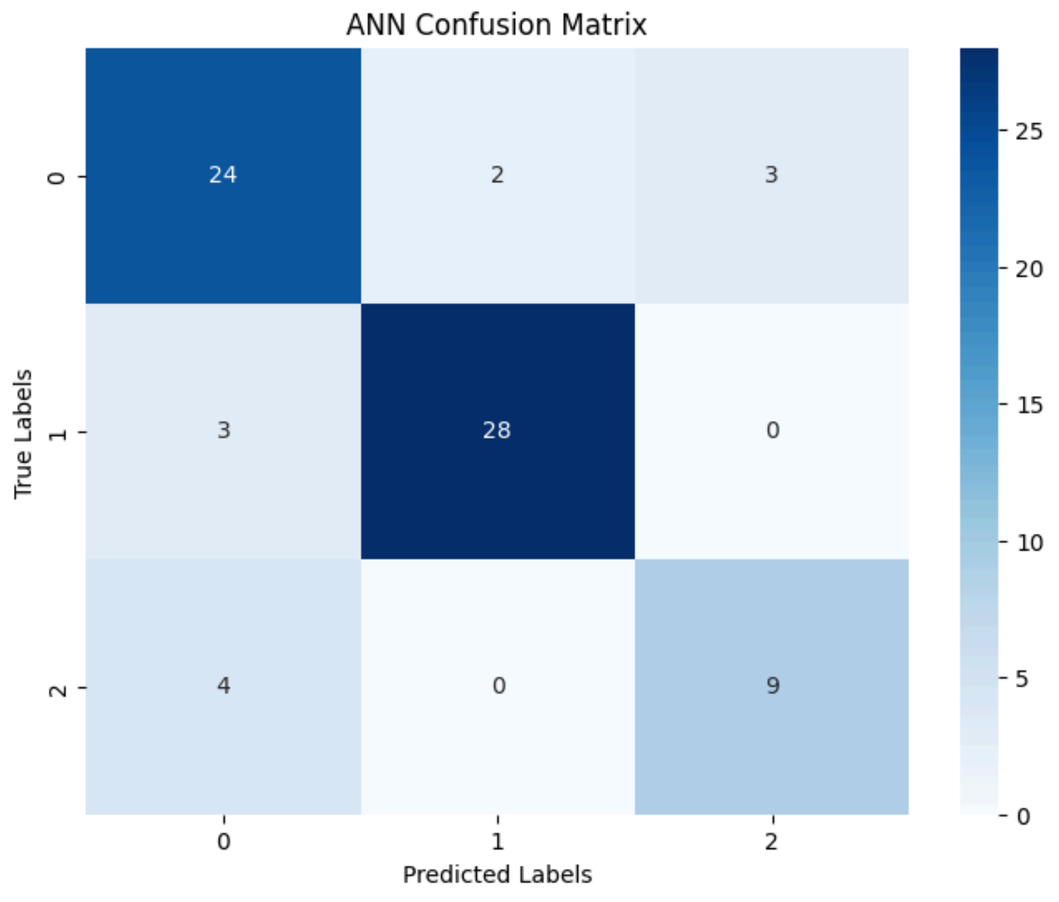
**ANN**
Two layered N. Nets.
Stochastic Gradient Descent (SGD).
Training over 20,000 epochs.
Feature scaling for improved performance.

**Optimization Approaches**
Learning rate
Feature scaling impact
Class balance
Epochs

# Results

| Metric | Logistic Regression | ANN |
|---|---|---|
| Accuracy | 83% | 92% |



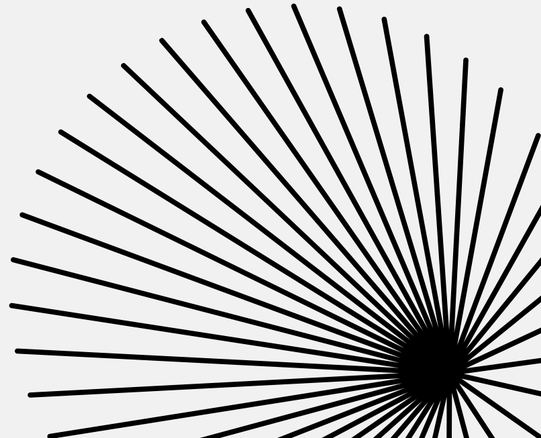
# Conclusion

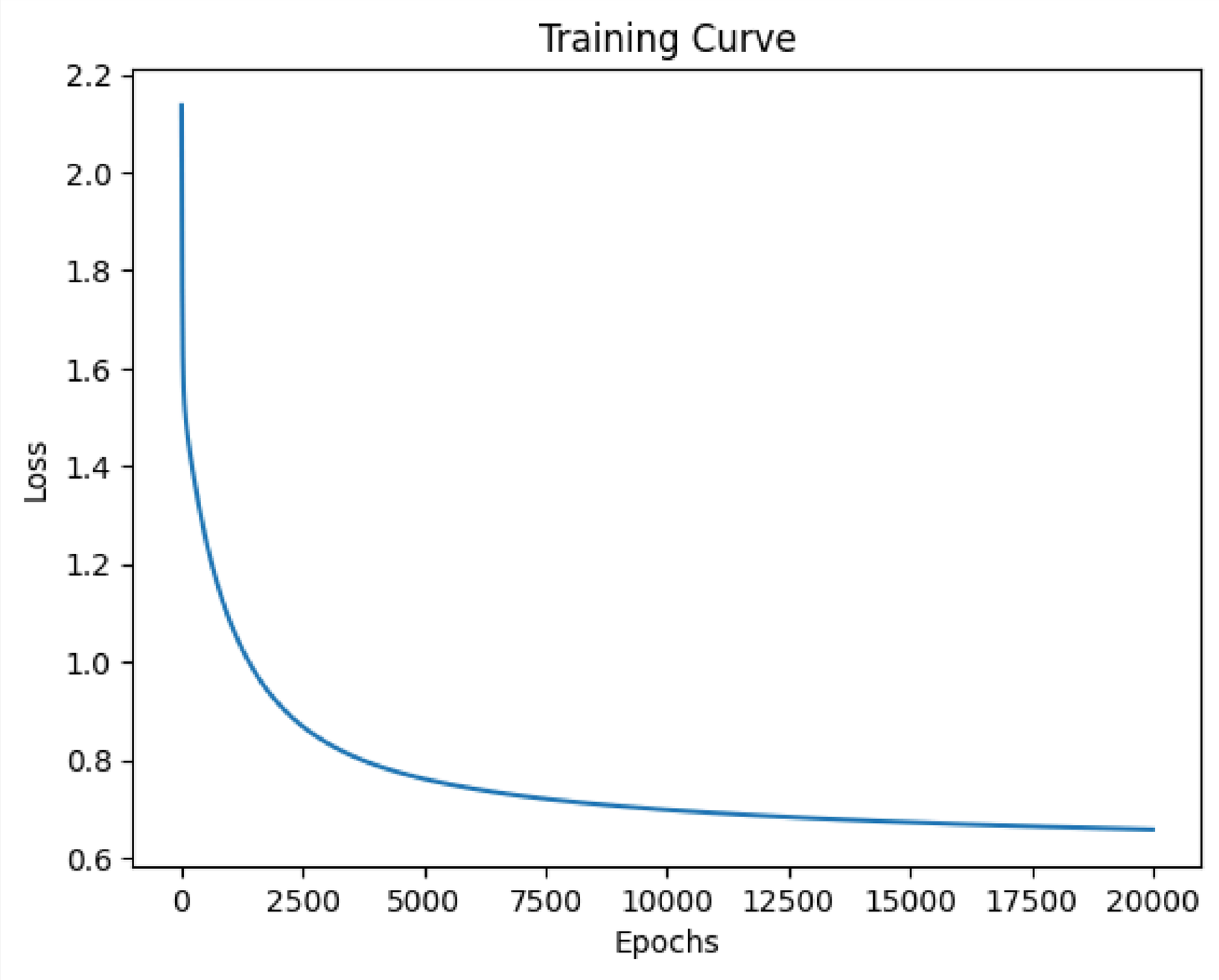
The project presented a survival analysis of patients diagnosed with (PBC), utilizing data from a Mayo Clinic study.  The goal was to predict patient outcomes using multi-class logistic regression and artificial neural network techniques. The dataset was preprocessed for missing values, class imbalance, and outliers, followed by modeling with logistic regression and artificial neural networks.

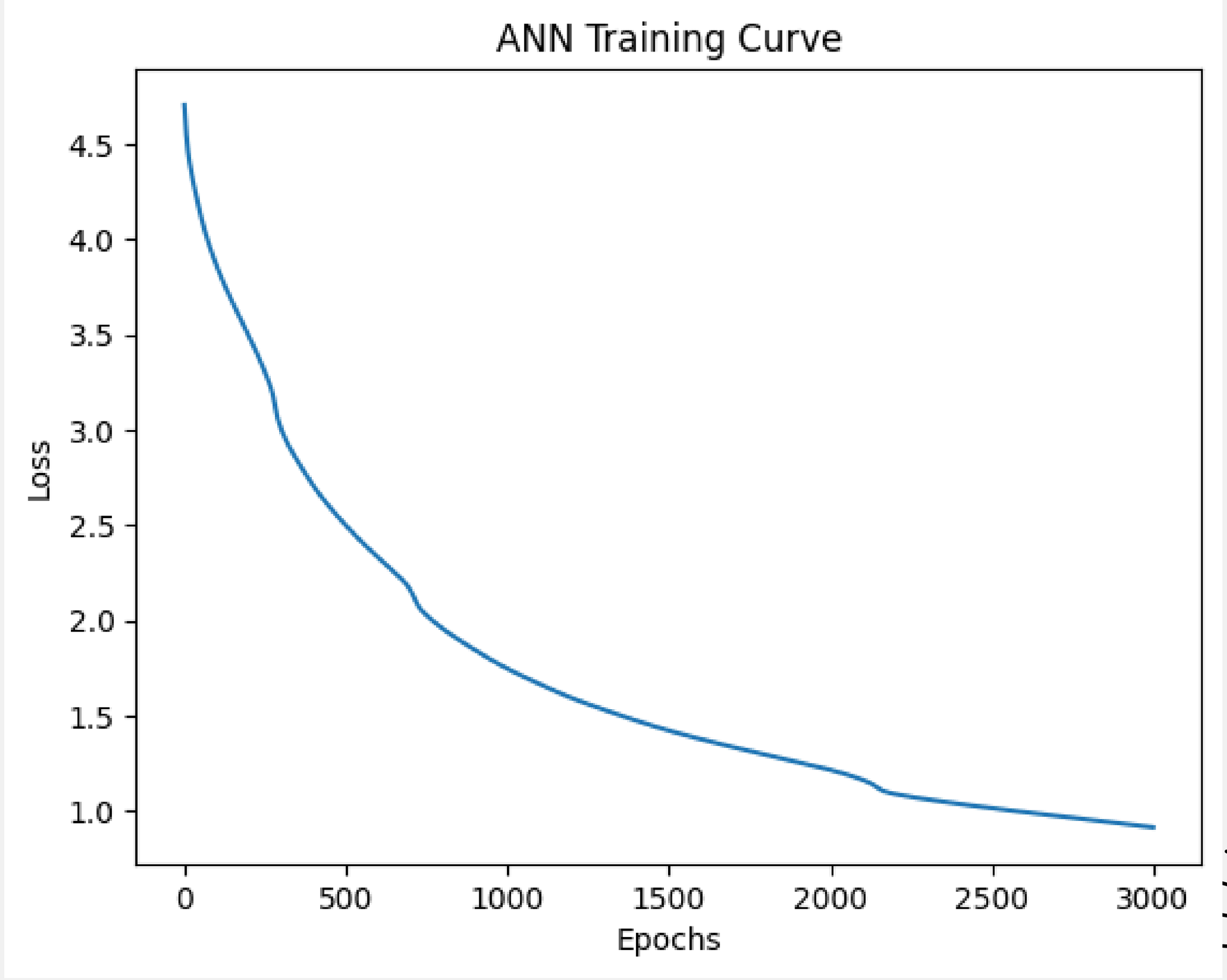
The artificial neural network achieved a higher accuracy (92%) compared to logistic regression (83%).

Current work aimed to provide insights on patients status based on clinical data and treatment history, potentially aiding in better clinical decision

# THANK YOU




Training Curve
Loss
2.2
2.0
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0
2500
5000
7500
10000
12500
15000
17500
20000
Epochs


ANN Training Curve
Loss
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0
500
1000
1500
2000
2500
3000
Epochs

# Part B

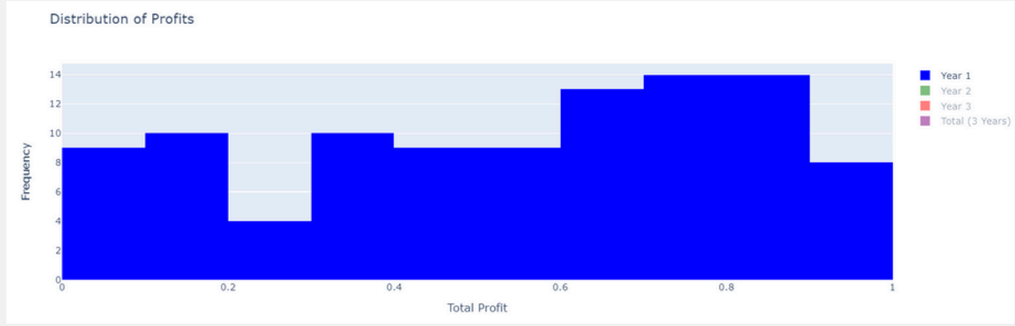
## Three year simulation model

```python
def simulate_and_plot(num_simulations=100):
    np.random.seed(42)

    # Initialize lists for costs, revenues, and profits
    year_costs, year_revenues, year_profits = [[] for _ in range(3)], [[] for _ in range(3)], [[] for _ in range(3)]

    # Simulation loop
    for _ in range(num_simulations):
        sim_results = three_year_sim()
        for i in range(3):
            year_costs[i].append(sim_results[i])
            year_revenues[i].append(sim_results[i + 3])
            year_profits[i].append(sim_results[i + 6])

    total_costs = sum(year_costs, [])
    total_revenues = sum(year_revenues, [])
    total_profits = sum(year_profits, [])

    # Prepare DataFrame for results
    results_df = pd.DataFrame(columns=['Year', 'Mean Cost', 'Cost Std Dev', 'Mean Revenue', 'Revenue Std Dev', 'Mean Profit', 'Profit Std Dev'])

    # Colors for histograms
    colors = ['blue', 'green', 'red', 'purple']

    # Create a figure for interactive plotting
    fig = go.Figure()

    # Processing, plotting, and storing results for each year and total
    for i in range(3):
        mean_cost, std_cost = calculate_statistics(year_costs[i])
        mean_revenue, std_revenue = calculate_statistics(year_revenues[i])
        mean_profit, std_profit = calculate_statistics(year_profits[i])

        # Storing results in DataFrame
        results_df = results_df.append({
            'Year': f'Year {i + 1}',
            'Mean Cost': mean_cost,
            'Cost Std Dev': std_cost,
            'Mean Revenue': mean_revenue,
            'Revenue Std Dev': std_revenue,
            'Mean Profit': mean_profit,
            'Profit Std Dev': std_profit
        }, ignore_index=True)

        # Plotting
        fig.add_trace(go.Histogram(x=year_profits[i], name=f'Year {i + 1}', marker_color=colors[i]))

    # Statistics and plot for the total of three years
    mean_cost, std_cost = calculate_statistics(total_costs)
    mean_revenue, std_revenue = calculate_statistics(total_revenues)
```

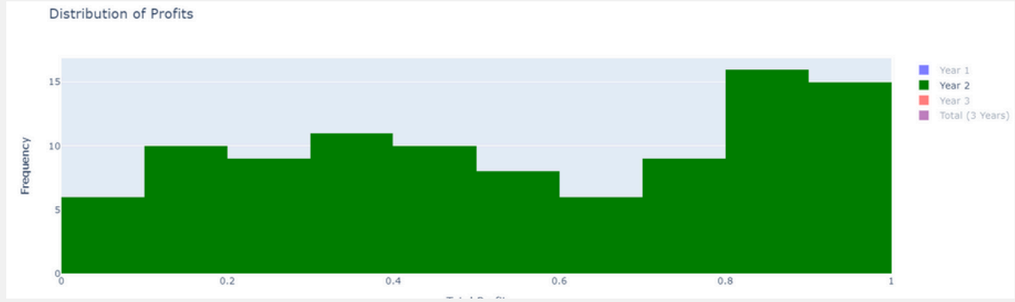
| | Year | Mean Cost | Cost Std Dev | Mean Revenue | Revenue Std Dev | Mean Profit | Profit Std Dev |
|---|---|---|---|---|---|---|---|
| 0 | Year 1 | 0.498202 | 0.293089 | 0.535892 | 0.265981 | 0.538401 | 0.285235 |
| 1 | Year 2 | 0.509265 | 0.276600 | 0.461908 | 0.320795 | 0.553123 | 0.294143 |
| 2 | Year 3 | 0.450492 | 0.299821 | 0.482464 | 0.292125 | 0.503528 | 0.296817 |
| 3 | Total (3 Years) | 0.485986 | 0.291120 | 0.493421 | 0.295471 | 0.531684 | 0.292846 |

Based on obtained results, bank should expect to make profit during first year. Profit of the second year will sightly increase compared to the first year. Bank will also make a profit in third year but slightly lower than the previous years (first and second). The mean profit and standard deviation over the three years implies that the bank's decision to migrate all potential clients is expected to be profitable on average over the three-year period, although there is some variability in profit from year to year.

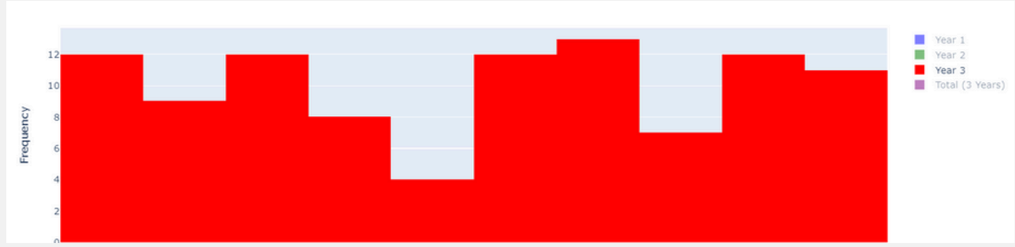
## YEAR 1




## YEAR 2




## YEAR 3




## CUMMULATIVE