# DS 340 Final Project Proposal:

# Customer Churn Prediction Using Artificial Neural Networks (ANN)

**Team:** Sungwoo Noh, Jaejoong Kim

**Elevator Pitch:** Promote an Artificial Neural Network (ANN)-based customer churn prediction model using a real-world banking dataset. We will predict whether a customer will leave the bank, enabling businesses to implement strategies to retain customers and reduce churn rate.

**Context:** According to Bain & Company, preventing 5% of customer churn can result in a 25-29% increase in operating profit. Understanding and predicting customer churn is crucial, especially in the competitive financial sector. Our project focuses on the Churn Modeling dataset, which includes customer behavior data from a bank. The goal is to utilize ANNs to detect complex patterns, improving predictive performance and actionable insights.

**Methods:**

1) **Exploratory Data Analysis (EDA):**
    a) Visualize the data to uncover trends in customer churn related to variables like 'Geography', 'Age', and 'CreditScore'.
    b) Identify any missing values or outliers.
2) **Data Preprocessing:**
    a) Encoding Categorical Data: Use one-hot encoding to convert categorical variables ('Geography', 'Gender') into numerical representations.
    b) Feature Scaling: Normalize numerical features ('Balance', 'EstimatedSalary') to help the ANN model converge efficiently.
    c) Addressing Imbalance: Apply SMOTE (Synthetic Minority Oversampling Technique) to ensure the model handles both churn and non-churn classes fairly.
3) **Building the ANN Model:**

a) Architecture: Input layer with 14 features, 2-3 hidden layers with ReLU activation, and a Sigmoid-activated output layer for binary classification.
b) **Training Parameters:**
    i) Loss Function: Binary Cross-Entropy
    ii) Optimizer: Adam optimizer for efficient learning
    iii) Batch size: 32, Epochs: 100

4) **Evaluation Metrics:**
    a) Use Accuracy, Precision, Recall, F1-Score, and Confusion Matrix to measure model performance.
    b) Comparison: We will compare the ANN model with Multiple Linear Regression, Logistic Regression, and Random Forest as baselines.

**Data Source:**

1) We will use the Churn Modeling dataset from Kaggle. The dataset contains 10,000 rows and 14 columns, with customer information such as 'CustomerId', 'CreditScore', 'Geography', 'Age', and the 'Exited' column.
2) **Dataset Link:** Churn Modelling Dataset

**Code Resources:**

1) **Libraries:**
    a) TensorFlow/Keras: For building and training the ANN model
    b) Pandas and NumPy: For data preprocessing
    c) Matplotlib and Seaborn: For visualizations
2) **ChatGPT:**
    a) Brainstorm solutions to encountered problems during development.
    b) Review and debug code snippets and optimize hyperparameter tuning strategies.
    c) Refine the project documentation to ensure clarity and completeness.

**What's New?**

1) **Customized ANN Architecture:**

a) We will modify the number of hidden layers and neurons to optimize model performance for this specific dataset.
b) We will explore alternative activation functions and evaluate their effect on the model's accuracy (e.g., using Leaky ReLU in hidden layers).

2) **Feature Engineering and Handling Class Imbalance:**
   a) Implement SMOTE to oversample the minority class (churned customers) and improve the model's ability to predict churn accurately.
   b) We will perform feature selection experiments to determine the most influential variables affecting churn.
3) **Hyperparameter Tuning:**
   a) We will perform grid search and random search to fine-tune the learning rate, batch size, and optimizer (e.g., testing between Adam and RMSprop).
   b) The goal is to find the configuration that maximizes the model's F1 score while minimizing overfitting.
4) **Additional Planned Efforts:**
   a) We will conduct model comparison experiments by implementing baseline models such as Multiple Linear Regression, Logistic Regression, and Random Forest to highlight the benefits of using ANN.
   b) Gather additional public datasets to validate our model's performance on other churn-related scenarios beyond the original Kaggle dataset.

**Plan and Milestones:**

1) **Milestone 1 (Nov 7):**
   a) Complete data preprocessing and exploratory analysis.
   b) Train the initial ANN model and evaluate its performance.
2) **Milestone 2 (Nov 30):**
   a) Perform hyperparameter tuning and finalize the ANN model.
   b) Complete the summary paper and finalize the presentation slides.
3) **Final Submission (Dec 9):**
   a) Deliver the final model, summary paper, and presentation slides.

**Proposed Demonstration or Evaluation:**

We will measure the model's success using accuracy, F1-score, and other metrics. Additionally, we will compare the performance of the ANN model with Multiple Linear Regression, Logistic Regression, and Random Forest models to highlight the benefits of using ANN for churn prediction.

**Experiments:**

1) **Feature Selection and Engineering**
    a) Select the most important features using forward selection and Lasso regularization.
    b) Compare the model performance when the model is built with all the features and the model is built with the selected features
    c) Evaluate the impact of feature selection on model training time and generalization based on its accuracy.
2) **Handling Imbalanced Data**
    a) Use SMOTE to oversample the minority class and compare model performance with and without SMOTE, focusing on recall for the churn class.
3) **Hypothesis testing**
    a) Through EDA, identify a feature that is likely to have a strong influence on churn.
    b) Construct the hypothesis that there is no difference in the likelihood of churn with and without the feature (ex. *has_credit_card)* whereas the alternative hypothesis is there is a significant difference in the likelihood of churn with the feature.
    c) Check the significance of the coefficient of the feature variable employing the Adjusted R-squared value to identify if the feature explains variation well in our dependent variable, *Exited*.
    d) For ANN, use SHAP(SHapley Additive exPlanations) values to see if having a credit card is an important factor in predicting churn to see if it supports our alternative hypothesis.
    e) Train two versions of the model where one model includes the features and the other model does not include the feature.

f) Perform Hypothesis Test on Model Performance using p-value.

**4) K-fold Cross Validation**

a) Split the dataset into K equal-sized folds (commonly K = 5 or 10).

b) Train and validate the model K times, using each fold as the validation set once and the remaining K-1 folds as the training set.

c) Record the metrics (e.g., accuracy, precision, recall, F1-score) for each fold.

d) Check if removing features (ex. *has_credit_card)* impacts the cross-validated accuracy, validating the hypothesis through consistent results.

e) Use cross-validation scores to ensure the model is not overfitting or underfitting.

f) Finalize the best model based on the average performance across folds and use it for further analysis or production deployment.