**Exercise: Building and Comparing ANN Models with ReLU and Leaky ReLU Activation Functions**

**Objective:**

The goal of this exercise is to build and compare the performance of two Artificial Neural Network (ANN) models: one using the ReLU activation function and the other using the Leaky ReLU activation function.

**Instructions:**

1. **Dataset Selection:**
   * Select a dataset from Kaggle suitable for binary classification tasks. Ensure that the dataset is balanced or take appropriate steps to handle any class imbalance if necessary.
2. **Data Preprocessing:**
   * Perform the necessary preprocessing steps on the dataset, such as handling missing values, feature scaling, and encoding categorical variables.
   * Split the dataset into training, validation, and test sets.
3. **Model Building:**
   * **Model 1: ReLU Network**
     + Build an ANN classification model using the ReLU activation function in the hidden layers.
     + Use a suitable architecture, optimizer, and loss function for binary classification.
     + Train the model on the training set and evaluate it on the validation set.
   * **Model 2: Leaky ReLU Network**
     + Build another ANN classification model using the Leaky ReLU activation function in the hidden layers.
     + Ensure the architecture, optimizer, and other hyperparameters are consistent with Model 1 for a fair comparison.
     + Train the model on the training set and evaluate it on the validation set.
4. **Model Comparison:**
   * Compare the performance of both models using metrics such as accuracy, precision, recall, F1-score, and AUC-ROC curve.
   * Visualize the training and validation loss/accuracy curves for both models.

**Exercise: Building and Comparing ANN Models with ReLU and PReLU Activation Functions**

**Objective:**

The goal of this exercise is to build and compare the performance of two Artificial Neural Network (ANN) models: one using the ReLU activation function and the other using the PReLU (Parametric ReLU) activation function.

**Instructions:**

1. **Dataset Selection:**
   * Select a dataset from Kaggle suitable for binary classification tasks. Ensure that the dataset is balanced or take appropriate steps to handle any class imbalance if necessary.
2. **Data Preprocessing:**
   * Perform the necessary preprocessing steps on the dataset, such as handling missing values, feature scaling, and encoding categorical variables.
   * Split the dataset into training, validation, and test sets.
3. **Model Building:**
   * **Model 1: ReLU Network**
     + Build an ANN classification model using the ReLU activation function in the hidden layers.
     + Use a suitable architecture, optimizer, and loss function for binary classification.
     + Train the model on the training set and evaluate it on the validation set.
   * **Model 2: PReLU Network**
     + Build another ANN classification model using the PReLU activation function in the hidden layers.
     + Ensure the architecture, optimizer, and other hyperparameters are consistent with Model 1 for a fair comparison.
     + Train the model on the training set and evaluate it on the validation set.
4. **Model Comparison:**
   * Compare the performance of both models using metrics such as accuracy, precision, recall, F1-score, and AUC-ROC curve.
   * Visualize the training and validation loss/accuracy curves for both models.

**Exercise: Building and Comparing ANN Models with ReLU and PReLU Activation Functions, Including PReLU Parameters and Learning Rate Adjustments with Nadam**

**Objective:**

The goal of this exercise is to build and compare the performance of two Artificial Neural Network (ANN) models: one using the ReLU activation function and the other using the PReLU (Parametric ReLU) activation function with parameter adjustments. Both models should use the Nadam optimizer with different learning rates.

**Instructions:**

1. **Dataset Selection:**
   * Select a dataset from Kaggle suitable for binary classification tasks. Ensure that the dataset is balanced or take appropriate steps to handle any class imbalance if necessary.
2. **Data Preprocessing:**
   * Perform the necessary preprocessing steps on the dataset, such as handling missing values, feature scaling, and encoding categorical variables.
   * Split the dataset into training, validation, and test sets.
3. **Model Building:**
   * **Model 1: ReLU Network**
     + Build an ANN classification model using the ReLU activation function in the hidden layers.
     + Use the Nadam optimizer with a learning rate of your choice (e.g., 0.001).
     + Define a suitable architecture, and use appropriate loss functions for binary classification.
     + Train the model on the training set and evaluate it on the validation set.
   * **Model 2: PReLU Network**
     + Build another ANN classification model using the PReLU activation function in the hidden layers.
     + Use the Nadam optimizer with a different learning rate (e.g., 0.01).
     + Ensure the architecture is consistent with Model 1 for a fair comparison.
     + Experiment with the parameters of the PReLU activation function (e.g., initial alpha values) and adjust them as needed.
     + Train the model on the training set and evaluate it on the validation set.
4. **Model Comparison:**
   * Compare the performance of both models using metrics such as accuracy, precision, recall, F1-score, and AUC-ROC curve.
   * Visualize the training and validation loss/accuracy curves for both models.

**Exercise: Building and Comparing ANN Models with ReLU and ELU Activation Functions**

**Objective:**

The goal of this exercise is to build and compare the performance of two Artificial Neural Network (ANN) models: one using the ReLU activation function and the other using the ELU (Exponential Linear Unit) activation function.

**Instructions:**

1. **Dataset Selection:**
   * Select a dataset from Kaggle suitable for binary classification tasks. Ensure that the dataset is balanced or take appropriate steps to handle any class imbalance if necessary.
2. **Data Preprocessing:**
   * Perform the necessary preprocessing steps on the dataset, such as handling missing values, feature scaling, and encoding categorical variables.
   * Split the dataset into training, validation, and test sets.
3. **Model Building:**
   * **Model 1: ReLU Network**
     + Build an ANN classification model using the ReLU activation function in the hidden layers.
     + Use a suitable architecture, optimizer, and loss function for binary classification.
     + Train the model on the training set and evaluate it on the validation set.
   * **Model 2: ELU Network**
     + Build another ANN classification model using the ELU activation function in the hidden layers.
     + Ensure the architecture, optimizer, and other hyperparameters are consistent with Model 1 for a fair comparison.
     + Train the model on the training set and evaluate it on the validation set.
4. **Model Comparison:**
   * Compare the performance of both models using metrics such as accuracy, precision, recall, F1-score, and AUC-ROC curve.
   * Visualize the training and validation loss/accuracy curves for both models.

**Exercise: Building and Comparing ANN Models with ReLU and SELU Activation Functions, with and without Dropouts**

**Objective:**

The goal of this exercise is to build and compare the performance of four different Artificial Neural Network (ANN) models using various activation functions and dropout techniques.

**Instructions:**

1. **Dataset Selection:**
   * Select a dataset from Kaggle suitable for binary classification tasks. Ensure that the dataset is balanced or take appropriate steps to handle any class imbalance if necessary.
2. **Data Preprocessing:**
   * Perform the necessary preprocessing steps on the dataset, such as handling missing values, feature scaling, and encoding categorical variables.
   * Split the dataset into training, validation, and test sets.
3. **Model Building:**
   * **Model 1: ReLU Network (No Dropout)**
     + Build an ANN classification model using the ReLU activation function in the hidden layers.
     + Do not use dropout layers.
     + Use a suitable architecture, optimizer, and loss function for binary classification.
     + Train the model on the training set and evaluate it on the validation set.
   * **Model 2: ReLU Network (With Dropout)**
     + Build another ANN classification model using the ReLU activation function in the hidden layers.
     + Include dropout layers to help prevent overfitting.
     + Ensure the dropout rate is set (e.g., 0.3 or 0.5).
     + Train the model on the training set and evaluate it on the validation set.
   * **Model 3: SELU Network (No Dropout)**
     + Build an ANN classification model using the SELU (Scaled Exponential Linear Unit) activation function in the hidden layers.
     + Do not use dropout layers.
     + Ensure to include SELU-specific initialization methods (e.g., Lecun Normal Initialization).
     + Train the model on the training set and evaluate it on the validation set.
   * **Model 4: SELU Network (With Dropout)**
     + Build another ANN classification model using the SELU activation function in the hidden layers.
     + Include dropout layers to help prevent overfitting.
     + Ensure the dropout rate is set (e.g., 0.3 or 0.5).
     + Train the model on the training set and evaluate it on the validation set.
4. **Model Comparison:**
   * Compare the performance of all four models using metrics such as accuracy, precision, recall, F1-score, and AUC-ROC curve.
   * Visualize the training and validation loss/accuracy curves for each model.

**Exercise: Building and Comparing ANN Models with ReLU and Its Variants**

**Objective:**

The goal of this exercise is to build and compare the performance of multiple Artificial Neural Network (ANN) models using different ReLU variants. You will evaluate the performance of these models and visualize the results using bar graphs.

**Instructions:**

1. **Dataset Selection:**
   * Select a dataset from Kaggle suitable for binary classification tasks. Ensure that the dataset is balanced or take appropriate steps to handle any class imbalance if necessary.
2. **Data Preprocessing:**
   * Perform the necessary preprocessing steps on the dataset, such as handling missing values, feature scaling, and encoding categorical variables.
   * Split the dataset into training, validation, and test sets.
3. **Model Building:**
   * Build the following ANN classification models using different ReLU variants:
     + **Model 1: Standard ReLU**
       - Use the ReLU activation function in the hidden layers.
       - Define a suitable architecture, optimizer, and loss function for binary classification.
       - Train the model and evaluate its performance on the validation set.
     + **Model 2: Leaky ReLU**
       - Use the Leaky ReLU activation function in the hidden layers.
       - Define a suitable architecture, optimizer, and loss function for binary classification.
       - Train the model and evaluate its performance on the validation set.
     + **Model 3: Parametric ReLU (PReLU)**
       - Use the PReLU activation function in the hidden layers.
       - Define a suitable architecture, optimizer, and loss function for binary classification.
       - Train the model and evaluate its performance on the validation set.
     + **Model 4: Exponential Linear Unit (ELU)**
       - Use the ELU activation function in the hidden layers.
       - Define a suitable architecture, optimizer, and loss function for binary classification.
       - Train the model and evaluate its performance on the validation set.
     + **Model 5: Scaled Exponential Linear Unit (SELU)**
       - Use the SELU activation function in the hidden layers.
       - Define a suitable architecture, optimizer, and loss function for binary classification.
       - Train the model and evaluate its performance on the validation set.
4. **Model Evaluation:**
   * For each model, record the final training and validation loss, as well as the accuracy.
5. **Visualization:**
   * Plot bar graphs to compare the final loss and accuracy for each model. Ensure that each graph clearly shows the performance of each ReLU variant.