AlphabetSoupCharity_Optimization

Explanation:

1. **Dropped More Columns:**
   - STATUS and SPECIAL_CONSIDERATIONS are removed. STATUS has very little variation (almost all values are the same), and SPECIAL_CONSIDERATIONS is likely redundant with other features or adds noise. Removing them simplifies the model.
2. **More Aggressive Binning:**
   - **APPLICATION_TYPE:** The cutoff was increased to 700 (from 500 in the original). This groups more rare application types into "Other".
   - **CLASSIFICATION:** The cutoff was increased to 1800 (from 1000). This further reduces the number of categories in the 'CLASSIFICATION' column.
3. **One-Hot Encoding (All Categorical Columns):**
   - Instead of creating a separate list, the pd.get_dummies() function is applied directly to the entire DataFrame. This ensures *all* categorical columns are encoded. This is cleaner and more robust.
4. **Added a Third Hidden Layer:**
   - A third hidden layer (hidden_nodes_layer3 = 20) with 'relu' activation is added. More layers can potentially capture more complex relationships in the data.
5. **Increased Neurons in Hidden Layers:**
   - The number of neurons in the first two hidden layers has been increased (hidden_nodes_layer1 = 100, hidden_nodes_layer2 = 50). More neurons per layer give the model more capacity to learn.
6. **Increased Epochs:**
   - The model is trained for 200 epochs (epochs=200). This allows the model more iterations to adjust its weights and potentially improve accuracy.
7. **Added a callback:**
   - Added a ModelCheckpoint callback to save the weights of the model during the training. This is set to save the model every 5 epochs. This is a good practice for saving model states.

- **Add Checkpoint Directory (Important!)**: TensorFlow needs a directory to save the checkpoints into. *Before* the training code. This code snippet does the following:
  * It imports the os library for creating the checkpoint file.
  * It creates the checkpoint folder. The filepath is set to save to this folder.
  * It uses os.makedirs(checkpoint_dir, exist_ok=True) to create the directory. The exist_ok=True part prevents an error if the directory already exists. This is crucial. If the directory doesn't exist, TensorFlow *will* raise an error and the model *won't* train.

- **Custom Callback:** The SaveEveryNepochs class is a *custom* Keras callback. This is the best way to handle saving every *n* epochs.

- **on_epoch_end:** The saving logic is now correctly placed inside the on_epoch_end method. This method is automatically called by Keras at the end of each epoch.

- **self.model:** Inside the callback, the model using self.model is accessed. This is how callbacks get a reference to the model they are attached to.

- **save_weights:** Critically, the code uses self.model.save_weights(path) to save *only* the weights. This is much more efficient and avoids errors.

- **File Extension:** .weights.h5 file extension *must* be used when saving *only* the model's weights using model.save_weights(). The callback enforces this to avoid confusion and potential errors.

- **Clear Output:** The code prints a message each time a checkpoint is saved, so you know it's working.

- **Directory Creation:** The critical directory creation step is included.

8. **Kept Good Practices from Original Starter Code:**

   - The code still preprocesses the data, splits it into training and testing sets, and scales the numerical features using StandardScaler.

   - The model still uses the adam optimizer and binary_crossentropy loss function, which are appropriate.

   - The model is still evaluated on the test set.