tarefa_1_tdnn_bloco_4

July 20, 2025

[21]:

```
from sklearn.model_selection import GridSearchCV
from sklearn.neural_network import MLPRegressor
from sklearn.preprocessing import MinMaxScaler, StandardScaler
from sklearn.pipeline import Pipeline
from sklearn import metrics
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from google import colab
colab.drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
```

[ ]:

```
df = pd.read_csv('/content/drive/MyDrive/POS AI/Perceptron_Datasets/
 ↪variacaoValores.csv')
df_test = df.iloc[100:120]
df = df.iloc[0:100]
```

[ ]:

```
def create_window_cols(df, max_window):
    df_new = pd.DataFrame()
    dados = df.iloc[:, 1].values
    tam = len(dados)

    for i in range(tam - max_window):
        row_data = {}
        for j in range(max_window + 1):
            row_data[f'x-{max_window - j}'] = dados[i + j]
        df_new_row = pd.DataFrame([row_data])
        df_new = pd.concat([df_new, df_new_row], ignore_index=True)

    y = df_new[['x-0']]
    x = df_new.drop('x-0', axis=1)

    return df_new, x, y
```

```
def execute_grid_search(x_train, y_train, x_test, y_test, hidden_layer_size,␣
 ↪grid_params):
    mlp_reg_rede1 = MLPRegressor(hidden_layer_sizes=hidden_layer_size)
    grid_rede1 = GridSearchCV(mlp_reg_rede1, grid_params, cv=5)
    grid_rede1.fit(x_train, y_train)
    y_predict = grid_rede1.predict(x_test)

    return y_predict, metrics.mean_absolute_error(y_test, y_predict), metrics.
 ↪mean_squared_error(y_test, y_predict), metrics.r2_score(y_test, y_predict), ␣
 ↪grid_rede1.best_estimator_.loss_, grid_rede1.best_estimator_.n_iter_
```

[ ]:

```
# Dados de treinamento
df_r1_train, x_r1_train, y_r1_train = create_window_cols(df, max_window=5)
df_r2_train, x_r2_train, y_r2_train = create_window_cols(df, max_window=10)
df_r3_train, x_r3_train, y_r3_train = create_window_cols(df, max_window=15)

# Dados de teste
df_r1_test, x_r1_test, y_r1_test = create_window_cols(df_test, max_window=5)
df_r2_test, x_r2_test, y_r2_test = create_window_cols(df_test, max_window=10)
df_r3_test, x_r3_test, y_r3_test = create_window_cols(df_test, max_window=15)
```

[ ]:

```
# Aplicação do scaler para os dados de teste
scaler = MinMaxScaler()

x_r1_train = scaler.fit_transform(x_r1_train)
x_r2_train = scaler.fit_transform(x_r2_train)
x_r3_train = scaler.fit_transform(x_r3_train)

y_r1_train = scaler.fit_transform(y_r1_train)
y_r2_train = scaler.fit_transform(y_r2_train)
y_r3_train = scaler.fit_transform(y_r3_train)
```

[ ]:

```
# Aplicação do scaler para os dados de teste
scaler = MinMaxScaler()

x_r1_test = scaler.fit_transform(x_r1_test)
x_r2_test = scaler.fit_transform(x_r2_test)
x_r3_test = scaler.fit_transform(x_r3_test)

y_r1_test = scaler.fit_transform(y_r1_test)
y_r2_test = scaler.fit_transform(y_r2_test)
y_r3_test = scaler.fit_transform(y_r3_test)
```

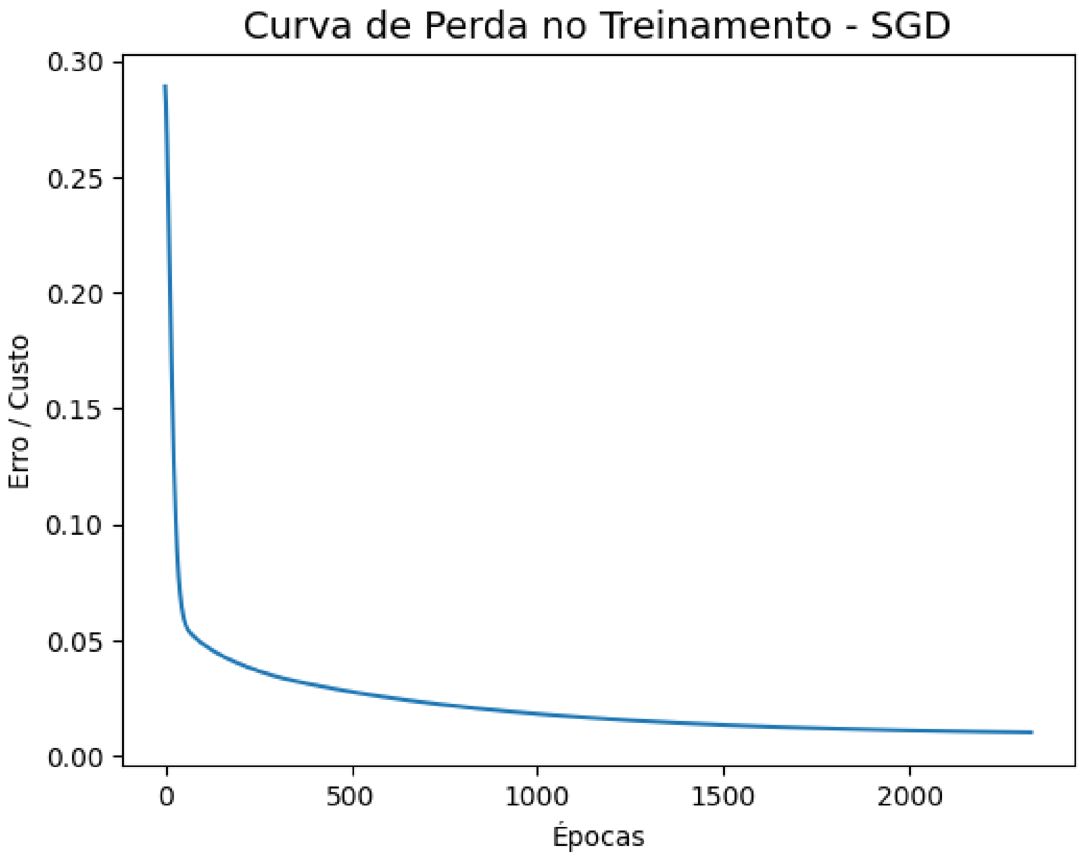
[ ]:

```
# Atividade 1 e 2 - Solver SGD
grid_params = {
    'hidden_layer_sizes': [(15)],
    'activation': ['relu'],
    'solver': ['sgd'],
    'max_iter': [5000],
    'tol': [0.000001],
    'momentum': [0.9],
    'early_stopping': [True],
    'epsilon': [1e-06],
    'learning_rate_init': [0.001],
    'learning_rate': ['constant']
}
mlp_reg_rede1 = MLPRegressor()
grid_rede1 = GridSearchCV(mlp_reg_rede1, grid_params, cv=5)
grid_rede1.fit(x_r1_train, y_r1_train)
y_predict = grid_rede1.predict(x_r1_test)

print("Score Rede1- Solver SGD: ", grid_rede1.best_score_)
print('Epochs: ', grid_rede1.best_estimator_.n_iter_)
print('Loss: ', grid_rede1.best_estimator_.loss_)
print('Mean Absolute Error (MAE): ', metrics.mean_absolute_error(y_r1_test,␣
 ↪y_predict))
print('Mean Square Error (MSE): ', metrics.mean_squared_error(y_r1_test,␣
 ↪y_predict))
print('Root Mean Squared Error (RMSE): ', metrics.mean_squared_error(y_r1_test,␣
 ↪y_predict))
print('R2 Score: ', metrics.r2_score(y_r1_test, y_predict))

plt.plot(grid_rede1.best_estimator_.loss_curve_)
plt.title('Curva de Perda no Treinamento - SGD', fontsize=14)
plt.xlabel('Épocas')
plt.ylabel('Erro / Custo')
plt.show()
```

```
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:691:
ConvergenceWarning: Stochastic Optimizer: Maximum iterations (5000) reached and
the optimization hasn't converged yet.
  warnings.warn(
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
```

```
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:691:
ConvergenceWarning: Stochastic Optimizer: Maximum iterations (5000) reached and
the optimization hasn't converged yet.
  warnings.warn(
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

```
Score Rede1- Solver SGD:  -0.0925541498535017
Epochs:  2327
Loss:  0.010084699924536658
Mean Absolute Error (MAE):  0.11804937659688576
Mean Square Error (MSE):  0.019637765510737735
Root Mean Squared Error (RMSE):  0.019637765510737735
R2 Score:  0.7292764568258777
```



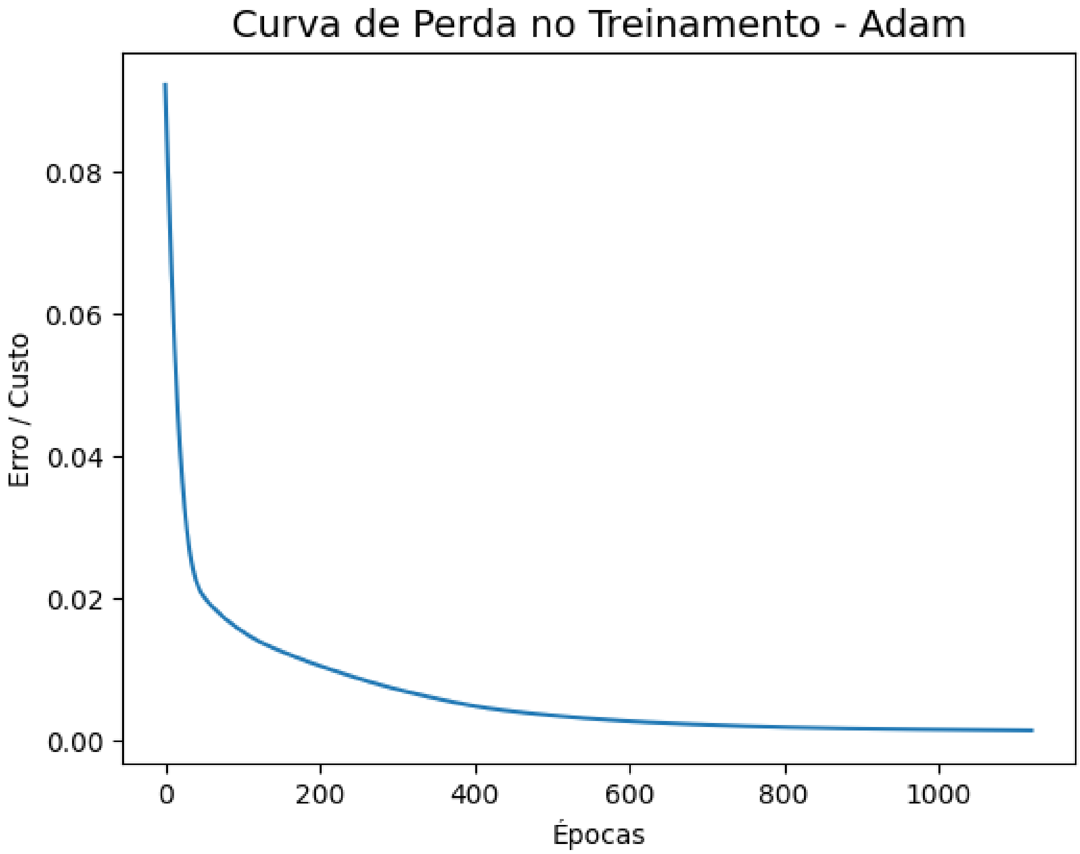
[ ]:

```
# Atividade 1 e 2 - Solver Adam
grid_params = {
    'hidden_layer_sizes': [(15)],
    'activation': ['relu'],
    'solver': ['adam'],
    'max_iter': [5000],
    'tol': [0.000001],
    'momentum': [0.9],
    'early_stopping': [True],
    'epsilon': [1e-06],
    'learning_rate_init': [0.001],
    'learning_rate': ['constant']
}
mlp_reg_rede1 = MLPRegressor()
grid_rede1 = GridSearchCV(mlp_reg_rede1, grid_params, cv=5)
grid_rede1.fit(x_r1_train, y_r1_train)
y_predict = grid_rede1.predict(x_r1_test)

print("Score Rede1- Solver Adam: ", grid_rede1.best_score_)
print('Epochs: ', grid_rede1.best_estimator_.n_iter_)
print('Loss: ', grid_rede1.best_estimator_.loss_)
print('Mean Absolute Error (MAE): ', metrics.mean_absolute_error(y_r1_test,␣
 ↪y_predict))
print('Mean Square Error (MSE): ', metrics.mean_squared_error(y_r1_test,␣
 ↪y_predict))
print('Root Mean Squared Error (RMSE): ', metrics.mean_squared_error(y_r1_test,␣
 ↪y_predict))
print('R2 Score: ', metrics.r2_score(y_r1_test, y_predict))

plt.plot(grid_rede1.best_estimator_.loss_curve_)
plt.title('Curva de Perda no Treinamento - Adam', fontsize=14)
plt.xlabel('Épocas')
plt.ylabel('Erro / Custo')
plt.show()
```

```
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
```

```
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

```
Score Rede1- Solver Adam:  -0.22778671676988188
Epochs:  1121
Loss:  0.001440330464322869
Mean Absolute Error (MAE):  0.051665171180737525
Mean Square Error (MSE):  0.004583213688079639
Root Mean Squared Error (RMSE):  0.004583213688079639
R2 Score:  0.9368164444125474
```



```
[ ]: # Atividade 3
     df = pd.read_csv('/content/drive/MyDrive/POS AI/Perceptron_Datasets/
      ↪variacaoValores.csv')

     # Dados de teste
     df_r1_test, x_r1_test, y_r1_test = create_window_cols(df.iloc[95:120],␣
      ↪max_window=5)
```

```
df_r2_test, x_r2_test, y_r2_test = create_window_cols(df.iloc[90:120],␣
 ↪max_window=10)
df_r3_test, x_r3_test, y_r3_test = create_window_cols(df.iloc[85:120],␣
 ↪max_window=15)

# Aplicação do scaler para os dados de teste
scaler = MinMaxScaler()

x_r1_test = scaler.fit_transform(x_r1_test)
x_r2_test = scaler.fit_transform(x_r2_test)
x_r3_test = scaler.fit_transform(x_r3_test)

y_r1_test = scaler.fit_transform(y_r1_test)
y_r2_test = scaler.fit_transform(y_r2_test)
y_r3_test = scaler.fit_transform(y_r3_test)

grid_params = {
    'activation': ['relu'],
    'solver': ['adam'],
    'max_iter': [5000],
    'tol': [0.000001],
    'momentum': [0.9],
    'early_stopping': [True],
    'epsilon': [1e-06],
    'learning_rate_init': [0.001],
    'learning_rate': ['constant']
}

df_results_atividade_3 = pd.DataFrame({'Treinamento': ['1º (T1)', '2º (T2)',␣
 ↪'3º (T3)']})
df_results_atividade_4 = pd.DataFrame({'Amostras': range(101, 121), 'f(t)':␣
 ↪df_test['f(t)']})


# --------------------------------
#             REDE 1
# --------------------------------
for i in range(3):
  predicts, mae, mse, r2, loss, epochs = execute_grid_search(x_r1_train,␣
 ↪y_r1_train, x_r1_test, y_r1_test, (15), grid_params)
  # Grid resultados Atividade 3
  df_results_atividade_3['Rede 1 - Perda T' + str(i)] = loss
  df_results_atividade_3['Rede 1 - Epochs T' + str(i)] = epochs

  # Grid resultados Atividade 4
  df_results_atividade_4['Rede 1 - T' + str(i)] = predicts
  df_results_atividade_4['Rede 1 - MAE T' + str(i)] = mae
  df_results_atividade_4['Rede 1 - MSE T' + str(i)] = mse
  df_results_atividade_4['Rede 1 - R2 Score T' + str(i)] = r2


# --------------------------------
#             REDE 2
# --------------------------------
for i in range(3):
  predicts, mae, mse, r2, loss, epochs = execute_grid_search(x_r2_train,␣
 ↪y_r2_train, x_r2_test, y_r2_test, (25), grid_params)
  # Grid resultados Atividade 3
  df_results_atividade_3['Rede 2 - Perda T' + str(i)] = loss
  df_results_atividade_3['Rede 2 - Epochs T' + str(i)] = epochs

  # Grid resultados Atividade 4
  df_results_atividade_4['Rede 2 - T' + str(i)] = predicts
  df_results_atividade_4['Rede 2 - MAE T' + str(i)] = mae
  df_results_atividade_4['Rede 2 - MSE T' + str(i)] = mse
  df_results_atividade_4['Rede 2 - R2 Score T' + str(i)] = r2


# --------------------------------
#             REDE 3
# --------------------------------
for i in range(3):
  predicts, mae, mse, r2, loss, epochs = execute_grid_search(x_r3_train,␣
 ↪y_r3_train, x_r3_test, y_r3_test, (50), grid_params)
  # Grid resultados Atividade 3
  df_results_atividade_3['Rede 3 - Perda T' + str(i)] = loss
  df_results_atividade_3['Rede 3 - Epochs T' + str(i)] = epochs

  # Grid resultados Atividade 4
  df_results_atividade_4['Rede 3 - T' + str(i)] = predicts
  df_results_atividade_4['Rede 3 - MAE T' + str(i)] = mae
  df_results_atividade_4['Rede 3 - MSE T' + str(i)] = mse
  df_results_atividade_4['Rede 3 - R2 Score T' + str(i)] = r2
```

```
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
```

```
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
```

```
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
```

```
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
```

```
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
```

```
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
```

```
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
```

```
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.11/dist-
packages/sklearn/neural_network/_multilayer_perceptron.py:1650:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

```
[ ]: display(df_results_atividade_3)
```

```
  Treinamento  Rede 1 - Perda T0  Rede 1 - Epochs T0  Rede 1 - Perda T1  \
0     1º (T1)           0.001596                1364           0.002782
1     2º (T2)           0.001596                1364           0.002782
2     3º (T3)           0.001596                1364           0.002782

   Rede 1 - Epochs T1  Rede 1 - Perda T2  Rede 1 - Epochs T2  \
0                1976             0.0012                 834
1                1976             0.0012                 834
2                1976             0.0012                 834

   Rede 2 - Perda T0  Rede 2 - Epochs T0  Rede 2 - Perda T1  \
0           0.000509                2055           0.000052
1           0.000509                2055           0.000052
2           0.000509                2055           0.000052

   Rede 2 - Epochs T1  Rede 2 - Perda T2  Rede 2 - Epochs T2  \
0                1813           0.000171                1904
1                1813           0.000171                1904
2                1813           0.000171                1904
```

```
   Rede 3 - Perda T0  Rede 3 - Epochs T0  Rede 3 - Perda T1  \
0           0.078984                  20           0.000034   
1           0.078984                  20           0.000034   
2           0.078984                  20           0.000034   

   Rede 3 - Epochs T1  Rede 3 - Perda T2  Rede 3 - Epochs T2  
0                1204           0.001875                 117  
1                1204           0.001875                 117  
2                1204           0.001875                 117  
```

[ ]:

```
from google.colab import sheets
sheet = sheets.InteractiveSheet(df=df_results_atividade_3)
```

[ ]:

```
display(df_results_atividade_4)
```

```
     Amostras    f(t)  Rede 1 - T0  Rede 1 - MAE T0  Rede 1 - MSE T0  \
100       101  0.4173     0.515729         0.057203         0.006123   
101       102  0.0062    -0.052466         0.057203         0.006123   
102       103  0.3387     0.664209         0.057203         0.006123   
103       104  0.1886     0.227142         0.057203         0.006123   
104       105  0.7418     0.824339         0.057203         0.006123   
105       106  0.3138     0.305637         0.057203         0.006123   
106       107  0.4466     0.526491         0.057203         0.006123   
107       108  0.0835     0.060664         0.057203         0.006123   
108       109  0.1930     0.209362         0.057203         0.006123   
109       110  0.3807     0.356324         0.057203         0.006123   
110       111  0.5438     0.691652         0.057203         0.006123   
111       112  0.5897     0.741905         0.057203         0.006123   
112       113  0.3536     0.407852         0.057203         0.006123   
113       114  0.2210     0.232622         0.057203         0.006123   
114       115  0.0631     0.097666         0.057203         0.006123   
115       116  0.4499     0.561547         0.057203         0.006123   
116       117  0.2564     0.311992         0.057203         0.006123   
117       118  0.7642     1.014266         0.057203         0.006123   
118       119  0.1411     0.230597         0.057203         0.006123   
119       120  0.3626     0.439241         0.057203         0.006123   

     Rede 1 - R2 Score T0  Rede 1 - T1  Rede 1 - MAE T1  Rede 1 - MSE T1  \
100              0.915901     0.609794         0.071851         0.006954   
101              0.915901     0.012965         0.071851         0.006954   
102              0.915901     0.580530         0.071851         0.006954   
103              0.915901     0.093806         0.071851         0.006954   
104              0.915901     0.830361         0.071851         0.006954   
105              0.915901     0.350265         0.071851         0.006954   
106              0.915901     0.526493         0.071851         0.006954   
107              0.915901     0.050991         0.071851         0.006954   
108              0.915901     0.289483         0.071851         0.006954   
```

```
109              0.915901     0.393443         0.071851         0.006954
110              0.915901     0.579317         0.071851         0.006954
111              0.915901     0.724367         0.071851         0.006954
112              0.915901     0.366537         0.071851         0.006954
113              0.915901     0.220267         0.071851         0.006954
114              0.915901     0.122253         0.071851         0.006954
115              0.915901     0.486040         0.071851         0.006954
116              0.915901     0.416375         0.071851         0.006954
117              0.915901     0.986347         0.071851         0.006954
118              0.915901     0.143041         0.071851         0.006954
119              0.915901     0.458663         0.071851         0.006954

     Rede 1 - R2 Score T1  …  Rede 3 - MSE T0  Rede 3 - R2 Score T0  \
100              0.904491  …         0.185589             -1.548867
101              0.904491  …         0.185589             -1.548867
102              0.904491  …         0.185589             -1.548867
103              0.904491  …         0.185589             -1.548867
104              0.904491  …         0.185589             -1.548867
105              0.904491  …         0.185589             -1.548867
106              0.904491  …         0.185589             -1.548867
107              0.904491  …         0.185589             -1.548867
108              0.904491  …         0.185589             -1.548867
109              0.904491  …         0.185589             -1.548867
110              0.904491  …         0.185589             -1.548867
111              0.904491  …         0.185589             -1.548867
112              0.904491  …         0.185589             -1.548867
113              0.904491  …         0.185589             -1.548867
114              0.904491  …         0.185589             -1.548867
115              0.904491  …         0.185589             -1.548867
116              0.904491  …         0.185589             -1.548867
117              0.904491  …         0.185589             -1.548867
118              0.904491  …         0.185589             -1.548867
119              0.904491  …         0.185589             -1.548867

     Rede 3 - T1  Rede 3 - MAE T1  Rede 3 - MSE T1  Rede 3 - R2 Score T1  \
100     0.535314         0.019188         0.000512              0.992964
101    -0.004047         0.019188         0.000512              0.992964
102     0.433776         0.019188         0.000512              0.992964
103     0.212244         0.019188         0.000512              0.992964
104     0.940437         0.019188         0.000512              0.992964
105     0.378387         0.019188         0.000512              0.992964
106     0.552148         0.019188         0.000512              0.992964
107     0.096094         0.019188         0.000512              0.992964
108     0.244356         0.019188         0.000512              0.992964
109     0.502957         0.019188         0.000512              0.992964
110     0.685781         0.019188         0.000512              0.992964
111     0.732092         0.019188         0.000512              0.992964
112     0.431733         0.019188         0.000512              0.992964
```

```
113     0.263638          0.019188          0.000512              0.992964
114     0.078319          0.019188          0.000512              0.992964
115     0.553640          0.019188          0.000512              0.992964
116     0.319658          0.019188          0.000512              0.992964
117     0.959978          0.019188          0.000512              0.992964
118     0.158760          0.019188          0.000512              0.992964
119     0.446008          0.019188          0.000512              0.992964

     Rede 3 - T2  Rede 3 - MAE T2  Rede 3 - MSE T2  Rede 3 - R2 Score T2
100     0.598354           0.05488          0.004843              0.933489
101    -0.129920           0.05488          0.004843              0.933489
102     0.496284           0.05488          0.004843              0.933489
103     0.250123           0.05488          0.004843              0.933489
104     0.825014           0.05488          0.004843              0.933489
105     0.345182           0.05488          0.004843              0.933489
106     0.590619           0.05488          0.004843              0.933489
107     0.130614           0.05488          0.004843              0.933489
108     0.280937           0.05488          0.004843              0.933489
109     0.434309           0.05488          0.004843              0.933489
110     0.550609           0.05488          0.004843              0.933489
111     0.704326           0.05488          0.004843              0.933489
112     0.410126           0.05488          0.004843              0.933489
113     0.277963           0.05488          0.004843              0.933489
114     0.090222           0.05488          0.004843              0.933489
115     0.519086           0.05488          0.004843              0.933489
116     0.362526           0.05488          0.004843              0.933489
117     0.937688           0.05488          0.004843              0.933489
118     0.214400           0.05488          0.004843              0.933489
119     0.485877           0.05488          0.004843              0.933489

[20 rows x 38 columns]
```

[ ]:

```python
sheet = sheets.InteractiveSheet(df=df_results_atividade_4)
```

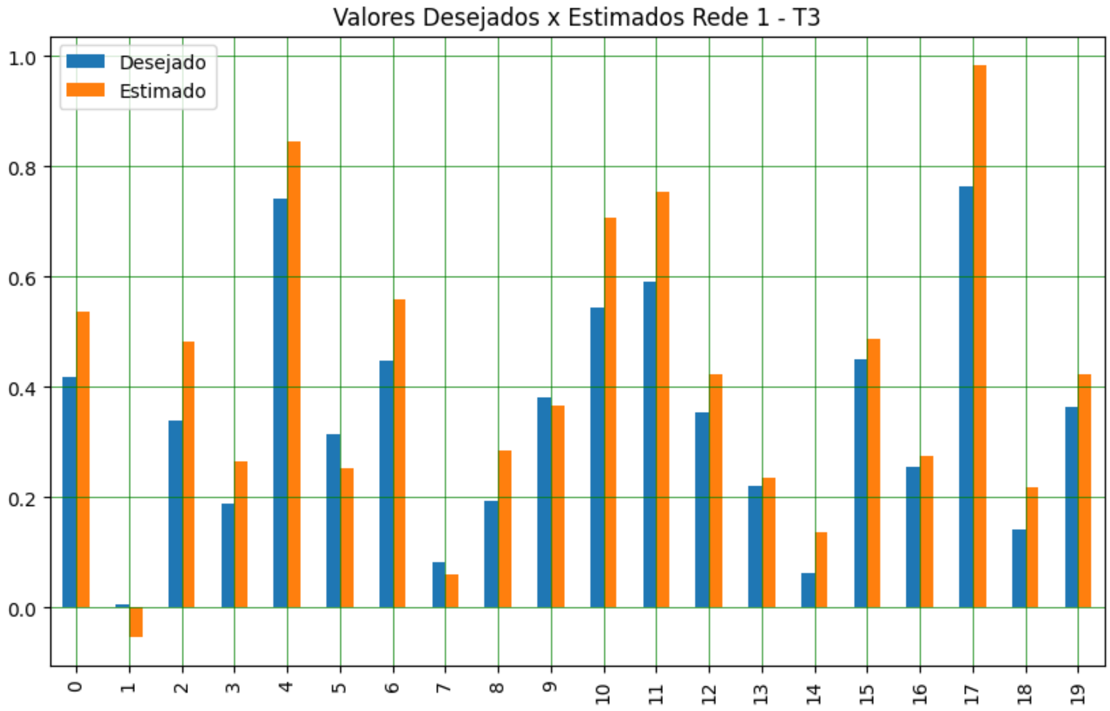
[30]:

```python
# csv com resultado em anexo nos arquivos da atividade
df = pd.read_csv('/content/drive/MyDrive/POS AI/Result_dataframes/Resultado␣
 ↪Tarefa bloco 4 Questão 4 - Página1.csv')
for col in df.columns:
    # Replace commas with periods in each string value
    df[col] = df[col].astype(str).str.replace(',', '.', regex=False)
    # Convert the column to numeric
    df[col] = pd.to_numeric(df[col])
```

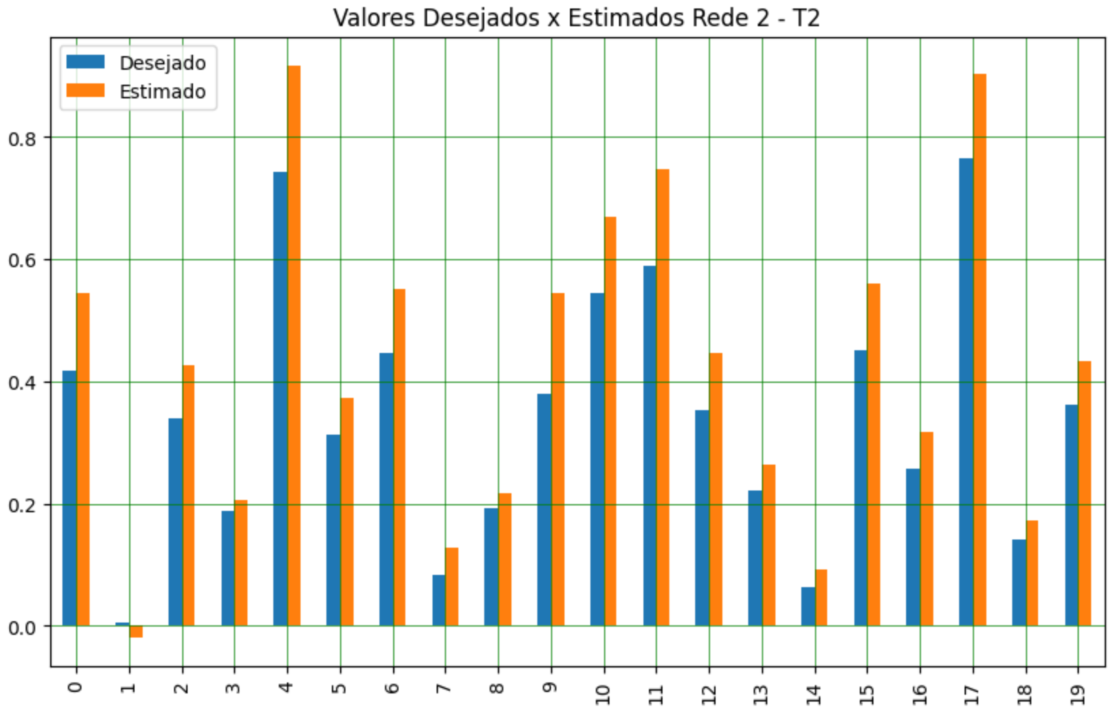
[32]:

```python
df_r1_predict = df[['f(t)', 'Rede 1 - T2']].rename(columns={'f(t)': 'Desejado',␣
 ↪'Rede 1 - T2': 'Estimado'})
df_r1_predict
df_r1_predict.plot(kind='bar',figsize=(10,6))
```

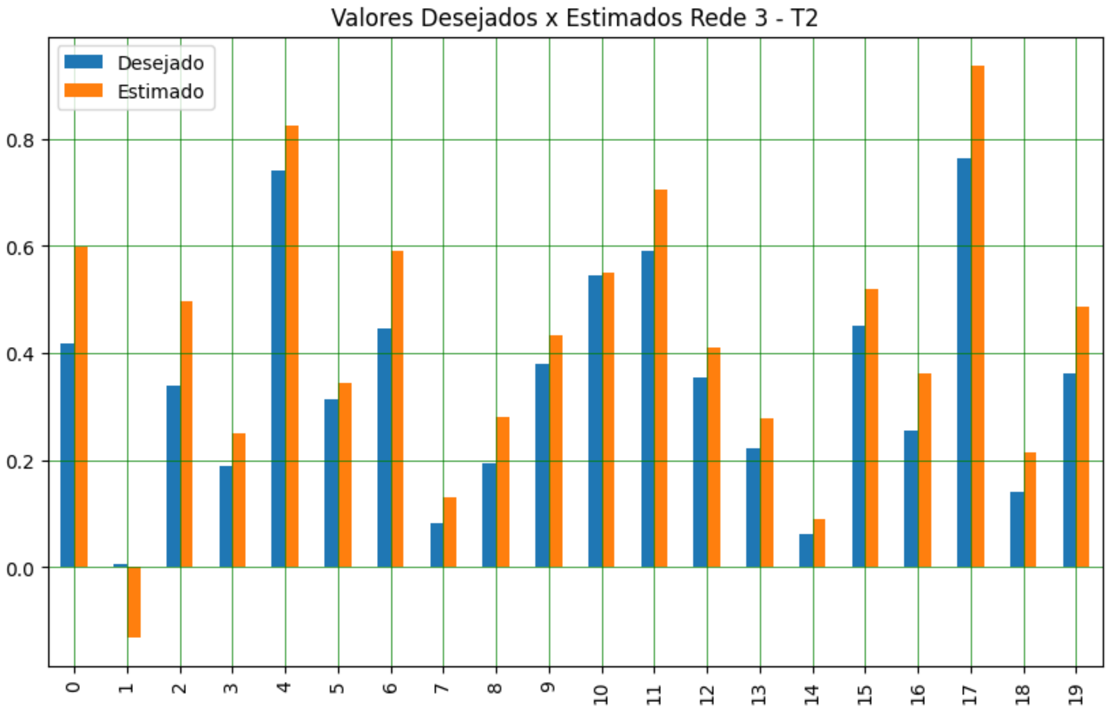
```
plt.title('Valores Desejados x Estimados Rede 1 - T3')
plt.grid(which='major', linestyle='-', linewidth='0.5', color='green')
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.show()
```



[33]:
```
df_r2_predict = df[['f(t)', 'Rede 2 - T2']].rename(columns={'f(t)': 'Desejado',␣
  ↪'Rede 2 - T2': 'Estimado'})
df_r2_predict
df_r2_predict.plot(kind='bar',figsize=(10,6))
plt.title('Valores Desejados x Estimados Rede 2 - T2')
plt.grid(which='major', linestyle='-', linewidth='0.5', color='green')
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.show()
```



```
[34]: df_r3_predict = df[['f(t)', 'Rede 3 - T2']].rename(columns={'f(t)': 'Desejado',␣
        ↪'Rede 3 - T2': 'Estimado'})
      df_r3_predict
      df_r3_predict.plot(kind='bar',figsize=(10,6))
      plt.title('Valores Desejados x Estimados Rede 3 - T2')
      plt.grid(which='major', linestyle='-', linewidth='0.5', color='green')
      plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
      plt.show()
```


Valores Desejados x Estimados Rede 3 - T2
Desejado
Estimado
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19