

# Mock Test Report

**Student:** Amit Kulkarni

**Date:** April 06, 2025



## Suggestions to Improve

The student demonstrates some understanding of machine learning concepts but has significant gaps in several key areas, particularly concerning data preprocessing, RNN architectures, time series forecasting, and imbalanced datasets.

Topic: Data Preprocessing and Scaling - Review the purpose and application of `StandardScaler` in `scikit-learn`. Practice scaling datasets with different characteristics.

Topic: Recurrent Neural Networks (RNNs) - Study the `return_sequences` argument in Keras RNN layers. Understand how it affects the output shape and its implications for building sequence-to-sequence and sequence-to-vector models.

Topic: Time Series Forecasting - Deepen your understanding of the limitations of simple moving averages, especially in the presence of noise. Explore alternative forecasting methods like ARIMA, exponential smoothing, or more advanced RNN architectures.

Topic: Handling Imbalanced Datasets - Review the SMOTE (Synthetic Minority Over-sampling Technique) algorithm. Understand how it works and its role in addressing class imbalance problems.

Topic: Regression Model Evaluation - Review common regression model evaluation metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R-squared, and Mean Absolute Error (MAE). Understand their strengths and weaknesses.

Topic: TensorFlow and Automatic Differentiation - Study the `tf.GradientTape()` context manager in TensorFlow. Understand its role in automatic differentiation and how it's used for calculating gradients during model training.

Topic: Reinforcement Learning - Review the concept of discounted rewards in reinforcement learning. Understand how the discount factor affects the value of future rewards and the overall learning process.

Focus on understanding the underlying mathematical principles of each ML concept.

Practice implementing the concepts using real-world datasets and different libraries (e.g., `scikit-learn`, TensorFlow, Keras).

Debug your code carefully and analyze the results to identify areas for improvement.

Consult relevant documentation and online resources to clarify any doubts.

## Correct Answers

Question	Correct Answer
What will be the output of the following code? <pre>python import numpy as np from sklearn.preprocessing import StandardScaler data = np.array([[1, 2], [3, 4], [5, 6]]) scaler = StandardScaler() scaled_data = scaler.fit_transform(data) print(scaled_data)</pre>	A NumPy array with each feature scaled independently to have zero mean and unit variance
Which statement best describes the purpose of the <code>`return_sequences=True`</code> argument in a Keras RNN layer?	It outputs a hidden state for each time step in the sequence.
Consider the following Python code using scikit-learn. What is the primary purpose of the code snippet? <pre>python from sklearn.model_selection import GridSearchCV from sklearn.ensemble import RandomForestClassifier param_grid = {'n_estimators': [50, 100, 200], 'max_depth': [None, 10, 20]} model = RandomForestClassifier() grid_search = GridSearchCV(model, param_grid, cv=5) grid_search.fit(X_train, y_train)</pre>	To perform hyperparameter tuning for a RandomForestClassifier using cross-validation.
What is the primary drawback of using a simple moving average for time series forecasting, especially with noisy data?	It lags behind significant changes in the data.
The following code snippet implements which reinforcement learning algorithm? <pre>python # ... (code for play_one_step, discount_rewards, etc. as provided in context) ... for episode in range(600): obs = env.reset() for step in range(200): epsilon = max(1 - episode / 500, 0.01) obs, reward, done, info = play_one_step(env, obs, epsilon) if done: break if episode &gt; 50: training_step(batch_size) ...</pre>	Deep Q-Network (DQN)
What does the following code snippet achieve in the context of handling imbalanced datasets? <pre>python from imblearn.over_sampling import SMOTE X_resampled, y_resampled = SMOTE().fit_resample(X_train, y_train)</pre>	It adds synthetic samples to the minority class.
Which of the following is NOT a common method for evaluating a regression model?	Accuracy
What is the primary purpose of the <code>`tf.GradientTape()`</code> context manager in TensorFlow?	To manage the computational graph for automatic differentiation.
In the context of RNNs for time series forecasting, what does a 'sequence-to-sequence' model predict compared to a 'sequence-to-vector' model?	A sequence of values representing future time steps.

The function `discount_rewards(rewards, discount_factor)` (as shown in the context) calculates what for each step in a reinforcement learning episode?

The sum of future discounted rewards.