Q1: Score distributions: Plot the genuine and impostor score distributions in a single plot.

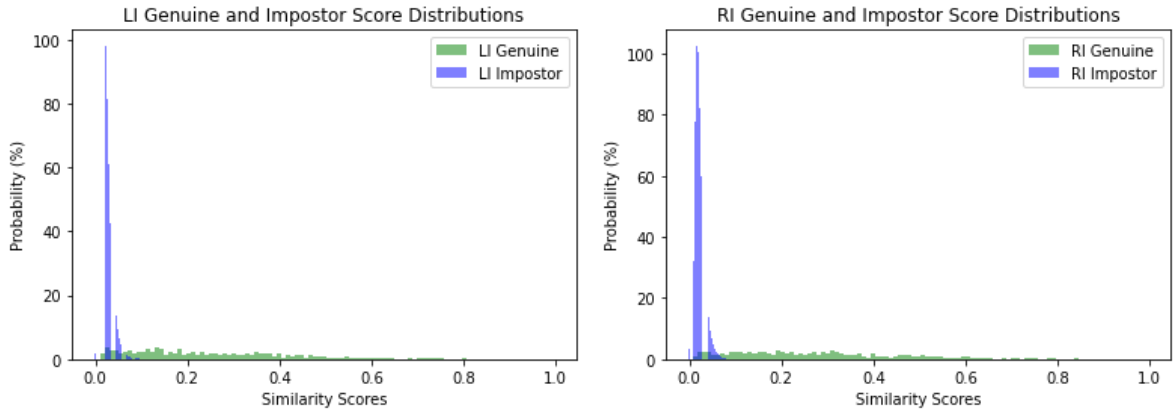


Figure.1

Do you need to normalize the distributions? Why (not)?

Normalization is not necessary for certain calculations, although it is useful for visualizing probability-similarity plots. This is because normalizing the genuine and impostor values can make them more similar and harder to distinguish, whereas the difference between these values can help to differentiate them. As you have observed, the density=True argument in the np.histogram() function normalizes the histogram so that the area under the curve sums to 1. This normalization is crucial for visualizing probability density distributions because the vertical axis represents the probability, and if I don't set density as True, it would show the number of samples on the vertical axis, which is hardly interpretable.

```
"""Plot the genuine and imposter score distributions."""
# Q1. Plot the genuine and impostor score distributions in a single plot.

def plot_score_distributions(scores, genuine_id, system_name):
    # Compute histograms of genuine and impostor scores

    genuine_hist, genuine_bins = np.histogram(scores[genuine_id == 1], bins=100, density=True)
    impostor_hist, impostor_bins = np.histogram(scores[genuine_id == 0], bins=100, density=True)

    # Plot histograms

    plt.bar(genuine_bins[:-1], genuine_hist, width=np.diff(genuine_bins), alpha=0.5, label=f'{system_name} Genuine', color='green')
    plt.bar(impostor_bins[:-1], impostor_hist, width=np.diff(impostor_bins), alpha=0.5, label=f'{system_name} Impostor', color='blue')

    plt.title(f'{system_name.upper()} Genuine and Impostor Score Distributions')
    plt.xlabel('Similarity Scores')
    plt.ylabel('Probability (%)')
    plt.legend()
    plt.show()

plot_score_distributions(li_scores, li_genuine_id, 'LI')
plot_score_distributions(ri_scores, ri_genuine_id, 'RI')
```

Describe qualitatively this combined plot (hint: limit the score range for better understanding)

The combined plot shows the genuine and impostor score distributions for two different biometric systems: the left fingerprints identification system (LI) and the right fingerprints identification system (RI). Each system has two histograms, one for genuine scores (scores from the same individual) and one for impostor

scores (scores from different individuals). Looking at the histograms of both systems, we can see that genuine scores (green) are shifted to the right meaning that has higher values than impostor scores (blue) for most of the score range, indicating that both systems are able to differentiate well between individuals and have higher similarity score for the same individual and lower for different ones.

Figure 2 shows the overlapping part of the figure1. As we can see, both systems can make a mistake in the overlapping area (approximately similarity score around 0.001 to 0.2).

Comparing the two systems, we can see that the RI system seems to have slightly better performance than the LI modality, as the genuine scores are generally higher and the impostor scores are generally lower and it has a less overlapping area between genuine and impostor. However, In the following parts of the report, I will show the Error calculation with different evaluation metrics and the two systems can be compared more concisely.

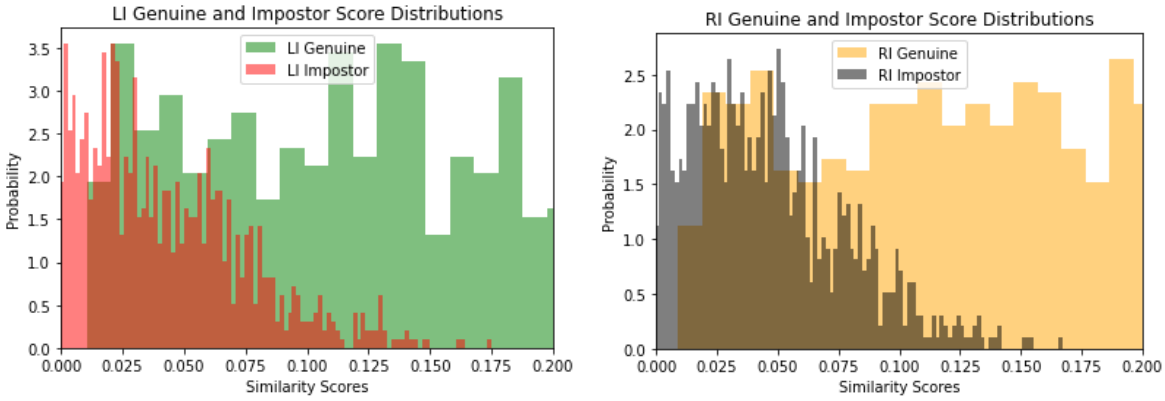


Figure.2

Q2: ROC Curves: Calculate FPR and TPR from the matching scores.

```
"""Calculate FPR, TPR from the matching scores."""
# Compute false positive rate and true positive rate
li_fpr, li_tpr, li_thresholds = roc_curve(li_genuine_id, li_scores)
ri_fpr, ri_tpr, ri_thresholds = roc_curve(ri_genuine_id, ri_scores)
```

Plot FAR and FRR as a function of matching scores.

As shown in Figure 3, there is a compromise between FAR and FRR, where improving one metric comes at the cost of degrading the other. The cross point of the FAR and FRR curve is called Equal Error Rate (EER). At this point, the system makes an equal number of false acceptances and false rejections.

In other words, the EER is the threshold where the biometric system achieves the optimal balance between security and convenience. If the threshold is set lower than the EER, the system becomes more convenient for users but less secure since more false acceptances occur. If the threshold is set higher than the EER, the system becomes more secure but less convenient for users since more genuine attempts are rejected. The

rejected area is the region to the left of the EER point on the ROC curve and the accepted area is the region to the right of it.

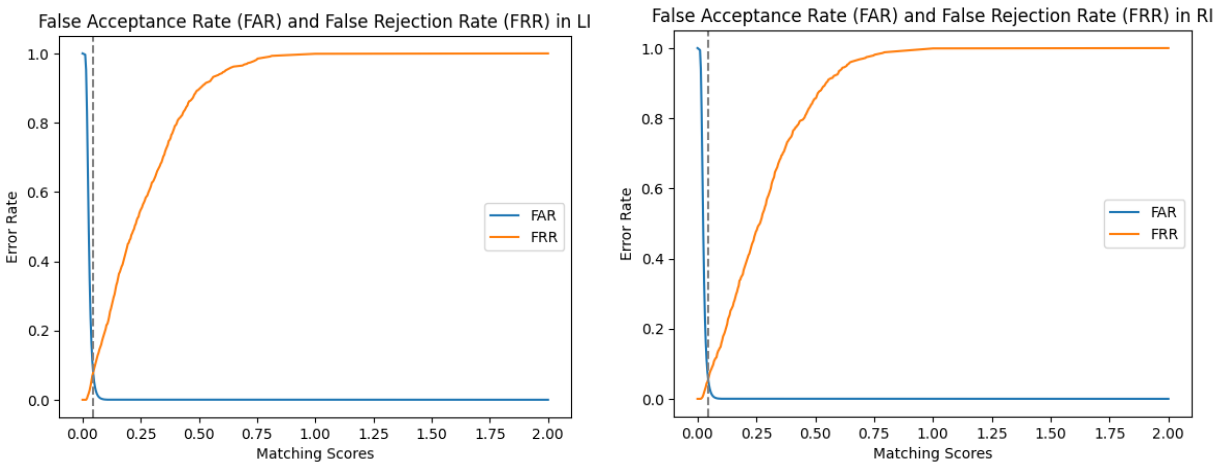


Figure.3

Figure 4 shows both systems have the same rejection area, however, RI has a slightly more accepted area.

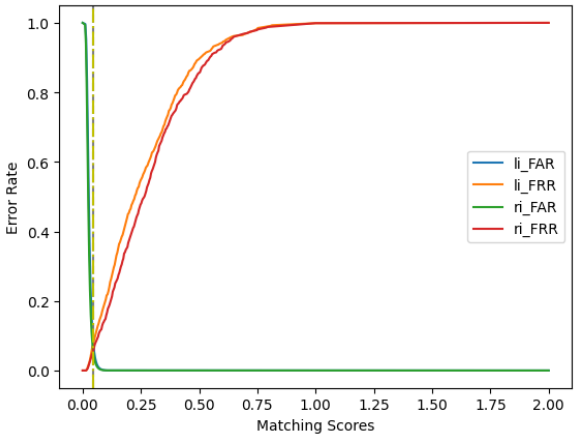


Figure.4

Plot the ROC curve. Plot for linear and logarithmic scale if needed. What do you observe?

As the left plot in figure 5 shows, both curves are above the diagonal, indicating that both systems perform better than random guessing. The performance of the LI system is slightly better than RI, as its ROC curve is always higher than the curve for LI. This difference in the right plot can be seen better because it is in logarithmic scale.

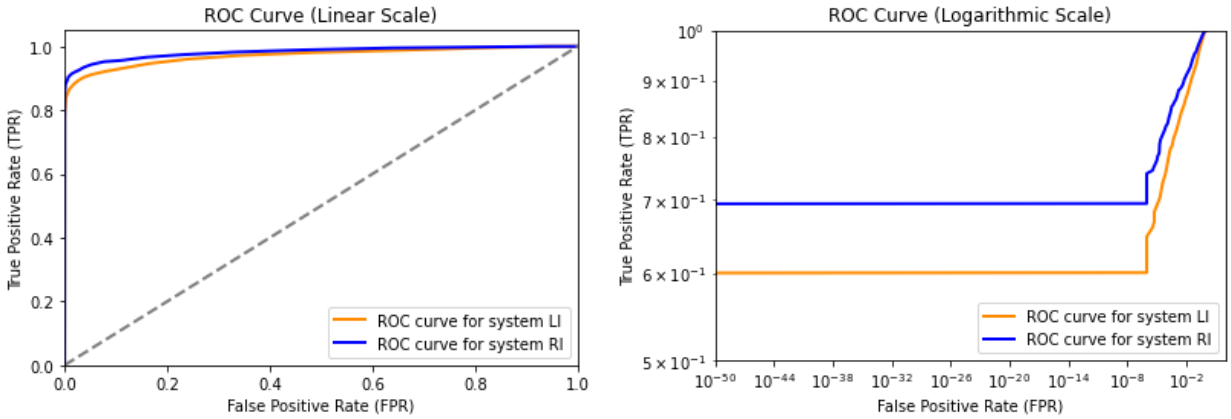


Figure.5

Plot the Detection Error Trade-off (DET) curve. How does it compare to ROC?

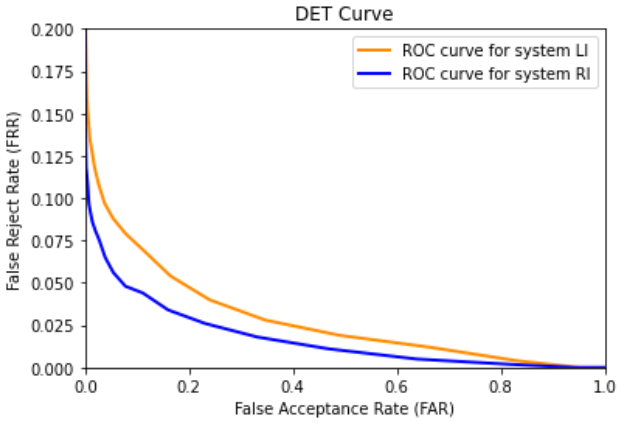


Figure.6

The ROC curve represents the relationship between the true positive rate (TPR) and false positive rate (FPR), with the ideal point being a TPR of one and an FPR of zero at the top left corner of the plot. DET curves are a variation of ROC curves where the False Negative Rate (FNR) is plotted on the y-axis, with the ideal point being the origin at the bottom left corner.

While ROC curves are plotted on a linear scale and classifiers often appear similar for most of the plot, as is also the case for LI and RI systems in figure 5, DET curves represent straight lines on a normal deviate scale, making it easier to visually assess the overall performance of different classification algorithms (figure 6).

DET curves provide direct feedback on the detection error tradeoff, allowing users to decide on the FNR they are willing to accept in exchange for the FPR or vice versa, aiding in operating point analysis. Therefore, the choice between ROC and DET curves depends on the specific needs of the system being evaluated and the evaluator's preferences.

Q3: Classification Metrics: Plot F1 and accuracy as a function of the decision thresholds on the similarity score.

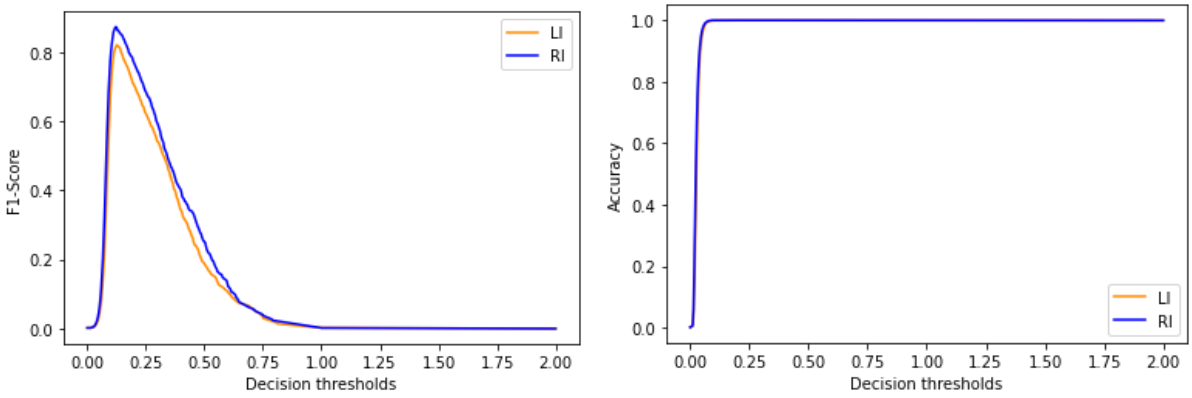


Figure.7

As you see both systems have exactly the same accuracy curve, while their F1-Score is different. Accuracy is heavily biased to the imbalance in the data and in our dataset, 10% of the data is an impostor and 90% is genuine. So, if the system just classifies the data as genuine without any process, it can achieve 90% accuracy! While F1-score is not influenced by the imbalance. It combines the precision and recall scores of the system and gives a better measure of the system's performance on both genuine and impostor attempts, regardless of the proportion of genuine and impostor attempts in the dataset. Therefore, the F1 score provides a more reliable measure of the system's overall performance compared to the accuracy score, especially when dealing with imbalanced datasets.

Calculate the threshold and accuracy for which F1 is maximal. Is it an interesting operating point?

For system LI, the maximum F1-score is 0.82, occurring at a threshold of 0.12. At this threshold, the corresponding accuracy is 0.99. For system RI, the maximum F1-score is 0.87, occurring at a threshold of 0.12. At this threshold, the corresponding accuracy is 0.99. So, as mentioned earlier, F1 score provides a more reliable measure of the system's overall performance

Do the same for the classification error (accuracy). Is there a difference? Is accuracy a good performance metric in this case?

System LI has a maximum accuracy of 0.9997 at threshold 0.13, with a corresponding F1-Score of 0.82, while System RI has a maximum accuracy of 0.9998 at threshold 0.12, with a corresponding F1-Score of 0.87. System RI has higher accuracy and F1-Score at its optimal threshold compared to System LI.

No, accuracy may not be a good performance metric in this case because the data seems to be imbalanced with a very high accuracy score and a very low F1 score. This suggests that the majority class is being correctly classified, while the minority class is being misclassified. In such cases, other evaluation metrics like F1 score, precision, recall, AUC-ROC, etc., should be used to get a more complete picture of the model's performance. These metrics take into account both false positives and false negatives, which is particularly important when dealing with imbalanced datasets.

Q4: AUC, EER, and alternatives: Calculate ROC AUC. Is this a good metric? What does it reveal about the system?

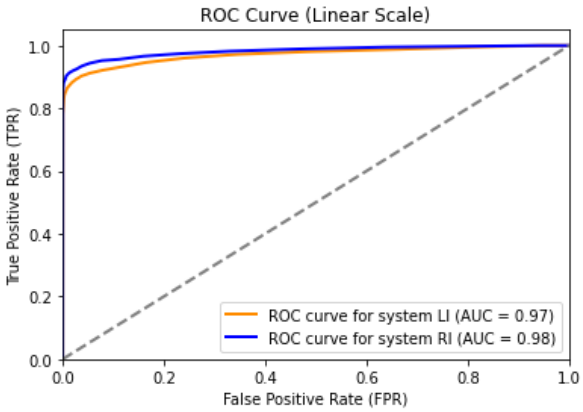


Figure.8

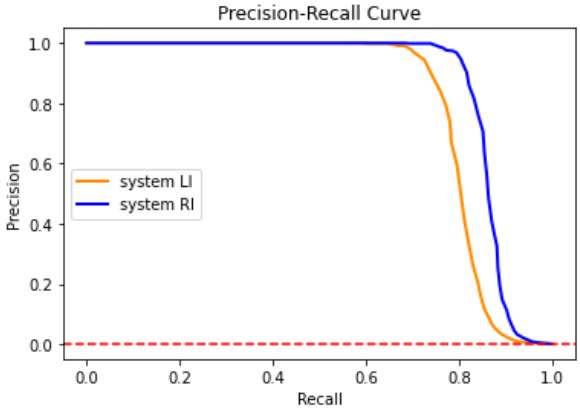


Figure.9

ROC AUC is a metric used to evaluate the ability of a binary classification model, including biometric systems, to distinguish between positive and negative samples across different decision thresholds. It ranges from 0 to 1, with higher values indicating better performance. ROC AUC provides a single-number evaluation of the system's ability to correctly identify genuine attempts and reject impostor attempts, regardless of the decision threshold. Therefore, it is a good metric to evaluate the performance of biometric systems. The ROC AUC value for system LI is 0.97 which is 1% less than the RI system.

Although the ROC AUC is a good metric to evaluate the performance of biometric systems, it does not provide information about the optimal decision threshold, which must be carefully selected based on the specific application requirements.

Calculate (by approximation) the EER and plot it on the FAR-FRR curve. Is this a good operation point?

Figure 10 shows the FAR-FRR curves and the EER point (the red point) for both systems. The EER is a good operating point for biometric systems, as it provides a balance between the system's ability to correctly identify genuine attempts and reject impostor attempts. However, the optimal operating point depends on the specific application requirements and the cost of false acceptance and false rejection errors. For example, in some fingerprint systems, the EER may not be a good operating point because the cost of false acceptance errors is higher than the cost of false rejection errors, and a more conservative operating point may be required. Other applications where the optimal operating point may differ from the EER include border control and national security systems, where a higher emphasis may be placed on the cost of false acceptance errors. EER value in the LI system is 0.92, while it is 0.94 in the RI system.

Calculate the decision threshold for which the sum of FRR and FAR is minimal.

In system LI, the minimal sum of FRR and FAR is 1 and it happens at a decision threshold of 0.198. In system RI, the minimal sum of FRR and FAR is 1.012 and it happens at a decision threshold of 0.22.

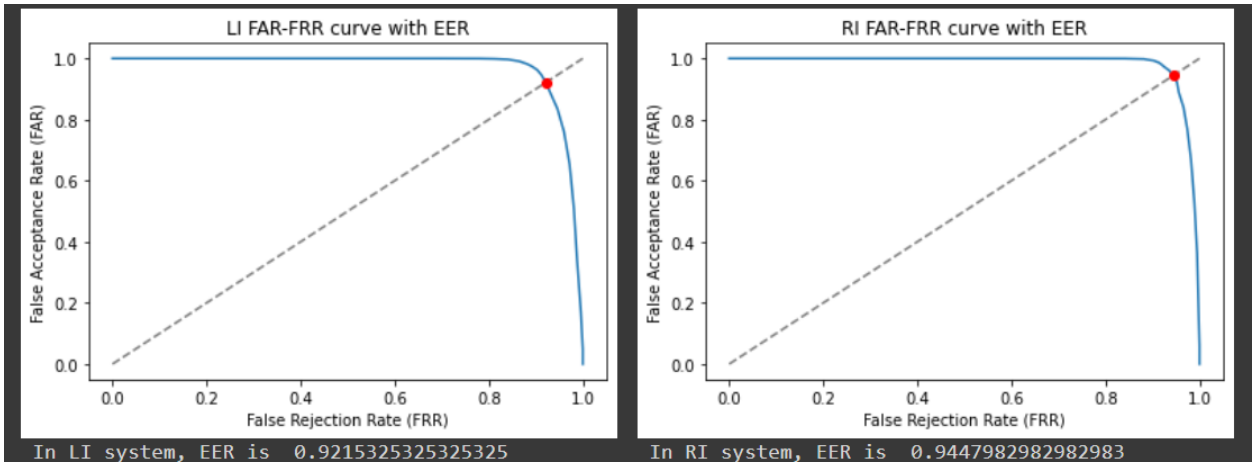


Figure.10

Is this point similar to the total classification error? Yes, the point where the sum of FRR and FAR is minimal is similar to the total classification error, as it represents the overall error rate of the biometric system. The total classification error can be calculated as (FRR + FAR) / 2

Can you suggest other strategies that give you an "optimal" performance? Calculate and discuss their (de)merits.

MCC is a binary classification metric that considers true/false positives and negatives, with a value between -1 to 1. It is useful for imbalanced data as it takes into account all confusion matrix elements. Using the Sklearn module, System LI has an MCC of 0.83 at a threshold of 0.12, while System RI has an MCC of 0.88 at the same threshold, indicating better performance. A difference of 0.05 in MCC is significant and can impact overall system performance.

Q5: Precision-Recall curves and related summary measures

Calculate and plot the Precision-Recall curve for this system. What does it reveal about the performance of the system?

As you can see in figure 9, the RI system has higher precision and recall values for most of the points on the precision-recall curve, indicating a better balance between precision and recall. And both models are above the baseline (the red dashed line) which is P / (P + N), indicating how much would the precision of a random classifier.

Calculate the Area Under the PR curve and the average precision scores. Discuss.

The area under the PR curve is a widely adopted metric for assessing the performance of binary classification models. A higher value is indicative of better performance. In this case, the value for RI (0.863) surpasses that of LI (0.803), suggesting that RI outperforms LI.

The value of the average precision scores provides a more detailed evaluation of the performance of a binary classification model across all possible recall levels. The score considers the trade-off between precision and recall and can be used to compare different models or settings for the same model. A high average precision

score indicates that the model has a high precision rate for all recall levels, which is desirable in most applications. Both LI and RI have a high area under the PR-curve values (0.877 and 0.889, respectively). This indicates that both models have high precision and recall for the fingerprint verification problem. However, the RI system is better than LI.

Q6: CMC curves: Calculate the Cumulative Matching Characteristic curve (implement this yourself)

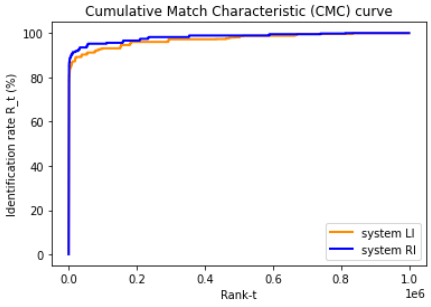
I plotted the CMC curve using a range of ranks from 0 to 1,000,000 (1 million) data points, which is why the identification rate reaches 100%.

```python
def calculate_cmc(scores, labels):
    # Sort the scores and labels in descending order
    sorted_indices = np.argsort(scores)[::-1]
    sorted_scores = scores[sorted_indices]
    sorted_labels = np.array(labels)[sorted_indices]

    # Calculate the number of genuine matches at each rank
    genuine_matches = np.cumsum(sorted_labels)

    # Calculate the rank-t identification rate R_t for t = 1, 2, ..., N
    identification_rates = (genuine_matches / np.sum(labels)) * 100

    return identification_rates
```



Compute the Rank-1 Recognition Rate.

It is 0.1 for both systems.

```python
li_identification_rates = calculate_cmc(li_scores, li_genuine_id)
ri_identification_rates = calculate_cmc(ri_scores, ri_genuine_id)

li_rank1_Recognition_rate = li_identification_rates[0]
ri_rank1_Recognition_rate = ri_identification_rates[0]
```

Q7: Evaluate different biometric systems

Use the above evaluation techniques to compare the biometric system based on the left and right indexes. Do you see any differences in any of the curves or measures?

For each evaluation metric, I compared the values for both systems. So, this question was answered in the above sections earlier.