Affective Computing - Project 1

Name: Rikshith Tirumanpuri
UID: U45426485

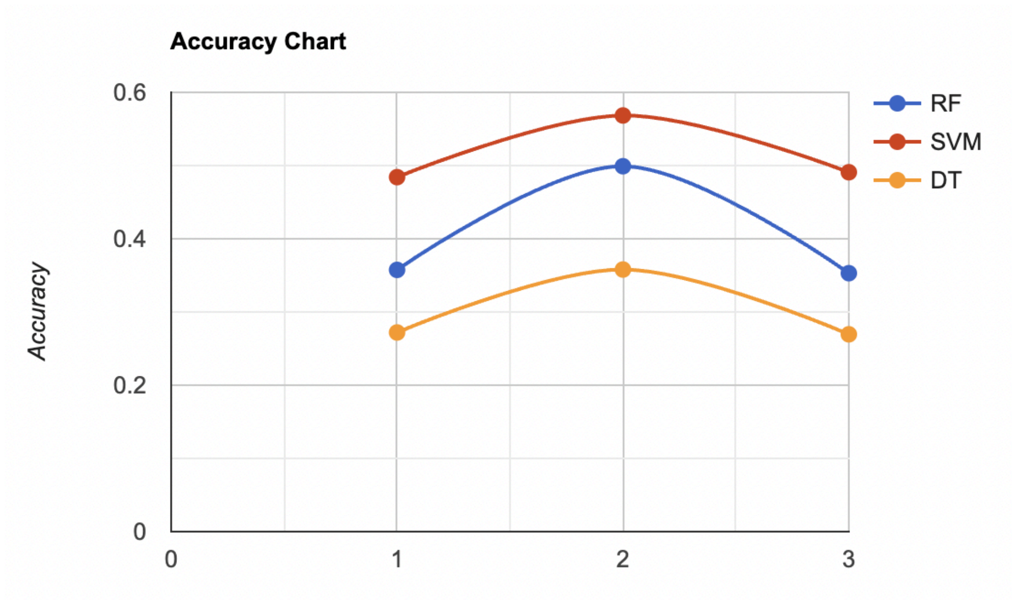
Report Questions

1. What are the classification results for each experiment?

Classifier / Data Type	Original			Data Translated			Rotated X | Y | Z		
Metric	**Accuracy**	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**
Random Forest	0.3575	0.3472	0.3583	0.4991	0.4851	0.4995	0.353	0.3434	0.3538
SVM	0.4842	0.5069	0.4845	0.5684	0.5905	0.5685	0.4909	0.5161	0.4913
Decision Tree	0.272	0.2662	0.2726	0.358	0.3552	0.3586	0.2695	0.2624	0.2702

2. Which of the classifiers worked the best for each data type (original, translated, rotated)? Why? If multiple had the same, then why did this happen? Note: this question is left intentionally vague. What does work best mean? Which metric is best? Think about what we've talked about in class in regards to expressions and emotion.

- Support Vector Machine (SVM) consistently outperformed other data types (original, translated, and rotated) in every scenario.
- It showed superior accuracy compared to techniques like Random Forest and Decision Tree.
- SVM's ability to determine optimal decision boundaries remained effective even with manipulated or varied data.
- Regardless of data complexities, SVM proved to be the best option for classification problems.
- The results suggest that SVM was the optimal choice for predicting outcomes in this case.



- As you can see from the line graph above, SVM performs better than the other two classifiers, with RF coming in second and Decision Trees coming in third.
- When we look at each one individually, we can see that translated data works better for all three classifiers (RF, SVM, and DT) than raw or rotated data.

3. For the top classifier, for each data type, describe the misclassification for each classifier. What was misclassified as what (e.g., sad looks like happy)? Based on your intuition about how expressions look, do the misclassifications make sense (i.e., do you think the expressions are similar – not based on any specific example)?

The top classifier according to my results is SVM.

The misclassifications for each label according to the data type are:

Original Data Confusion Matrix:
[[5388 2253 1355 901 3413 1003]
 [777 4106 1354 1045 286 870]
 [873 1547 3598 1700 1157 1526]
 [219 598 1456 5643 139 146]
 [2541 1030 1175 482 4991 883]
 [326 637 1106 202 156 5520]]

Insights:

- The incorrectly classified instances for the label “Angry” are 9925

- The incorrectly classified instances for the label “Disgust” are 4332
- The incorrectly classified instances for the label “Fear” are 6803
- The incorrectly classified instances for the label “Happy” are 3558
- The incorrectly classified instances for the label “Sad” are 6111
- The incorrectly classified instances for the label “Surprise” are 2427

Translated Data Confusion Matrix:

```
[[6473 2008 1203 1016 2722  936]
 [ 691 4993 1338  611  213  494]
 [ 552 1534 4196 1569  638 1317]
 [ 197  590 1462 6248   96  315]
 [2005  673 1136  444 6432  895]
 [ 206  373  709   85   41 5991]]
```

Insights:

- The incorrectly classified instances for the label “Angry” are 8885
- The incorrectly classified instances for the label “Disgust” are 3347
- The incorrectly classified instances for the label “Fear” are 5610
- The incorrectly classified instances for the label “Happy” are 2660
- The incorrectly classified instances for the label “Sad” are 5153
- The incorrectly classified instances for the label “Surprise” are 1414

Rotated X Data Confusion Matrix:

```
[[5523 2246 1413  926 3473 1033]
 [ 736 4154 1302  947  175  845]
 [ 860 1538 3612 1666 1143 1536]
 [ 186  601 1471 5720  151  129]
 [2521 1039 1139  506 5054  913]
 [ 298  593 1107  208  146 5492]]
```

Insights:

- The incorrectly classified instances for the label “Angry” are 9085
- The incorrectly classified instances for the label “Disgust” are 4005
- The incorrectly classified instances for the label “Fear” are 7743
- The incorrectly classified instances for the label “Happy” are 2538
- The incorrectly classified instances for the label “Sad” are 7118
- The incorrectly classified instances for the label “Surprise” are 2352

-

Rotated Y Data Confusion Matrix:

```
[[5525 2232 1456  932 3483 1053]
 [ 734 4164 1265  845  155  832]
 [ 871 1554 3662 1662 1140 1539]
 [ 183  640 1468 5759  145  125]
 [2516 1029 1114  564 5075  912]
 [ 295  552 1079  211  144 5487]]
```

Insights:

- The incorrectly classified instances for the label “Angry” are 9156
- The incorrectly classified instances for the label “Disgust” are 3831
- The incorrectly classified instances for the label “Fear” are 7766
- The incorrectly classified instances for the label “Happy” are 2561
- The incorrectly classified instances for the label “Sad” are 6135
- The incorrectly classified instances for the label “Surprise” are 2281

Rotated Z Data Confusion Matrix:

```
[[5531 2227 1463  933 3493 1060]
 [ 734 4165 1245  807  160  833]
 [ 881 1545 3700 1678 1112 1521]
 [ 175  649 1461 5773  153  129]
 [2510 1033 1117  574 5081  913]
 [ 293  552 1058  208  143 5492]]
```

Insights:

- The incorrectly classified instances for the label “Angry” are 9176
- The incorrectly classified instances for the label “Disgust” are 3779
- The incorrectly classified instances for the label “Fear” are 6737
- The incorrectly classified instances for the label “Happy” are 2567
- The incorrectly classified instances for the label “Sad” are 7147
- The incorrectly classified instances for the label “Surprise” are 2254

- Accurately judging the accuracy of emotion classification without direct access to the images is challenging.
- Facial expressions can contain overlapping features between different emotions, leading to potential misclassifications.

- Similarly, traits like patterns of noses and wide-opened eyeballs can be associated with both "Disgust" and "Fear," potentially leading to misclassification.
- It's important to recognize that the classifier's feature distinctions may not always align precisely with our intuitive understanding of emotional expression.
- Further research and investigation are necessary to refine and improve emotion classification algorithms and their accuracy.

4. Why do you think you got the results that you got for each of the different data types/classifiers (i.e., why are they different, or why are they the same)? For example, if SVM and RF have different results, why are they different? If they are the same – why are they the same?

- **Random Forest:**
- **Crucial Learning Point:** It does this by effectively handling missing data and performing well with large, complex datasets.
- **Observations:** With translated data, Random Forest performs noticeably better in this instance (accuracy increases from 0.3575 to 0.4991). This implies that the translated version of the data may have clarified some complexities present in the original data, enabling Random Forest to produce more accurate predictions. Its accuracy drops to 0.3530 when data is rotated, suggesting that it may be susceptible to specific data transformations.

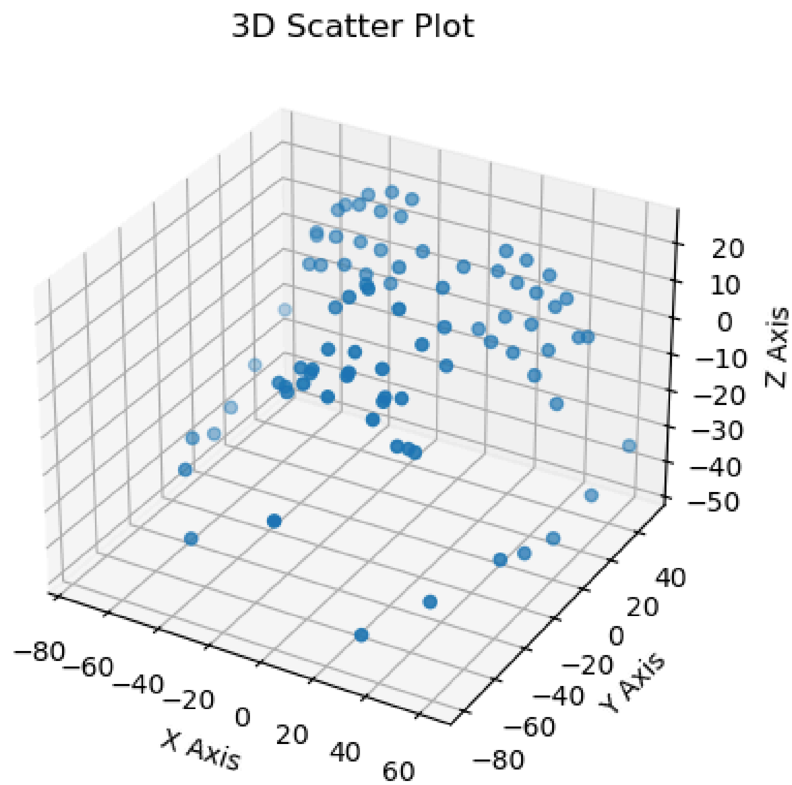
- **SVM (Support Vector Machine):**
- **Crucial Learning Point:** Strong at identifying hyperplanes that effectively divide classes; effective with high dimensional data**.**
- **Observations:** SVM consistently delivers good results with all kinds of data. It translates data with the highest accuracy (0.5684), indicating that it successfully preserves the differences between classes after translation. This is consistent with its ability to establish sensible boundaries for separation.

- **Decision Tree:**
- **Crucial Learning:** Simple to understand and effective for small datasets.
- **Observations:** Decision Tree performs the worst all around. In all cases, the accuracy stays comparatively constant (0.2720, 0.3580, 0.2695). This implies that the data may not have sufficient inherent complexity for the decision tree to learn from the variations, even after translation or rotation. Furthermore, it may be less

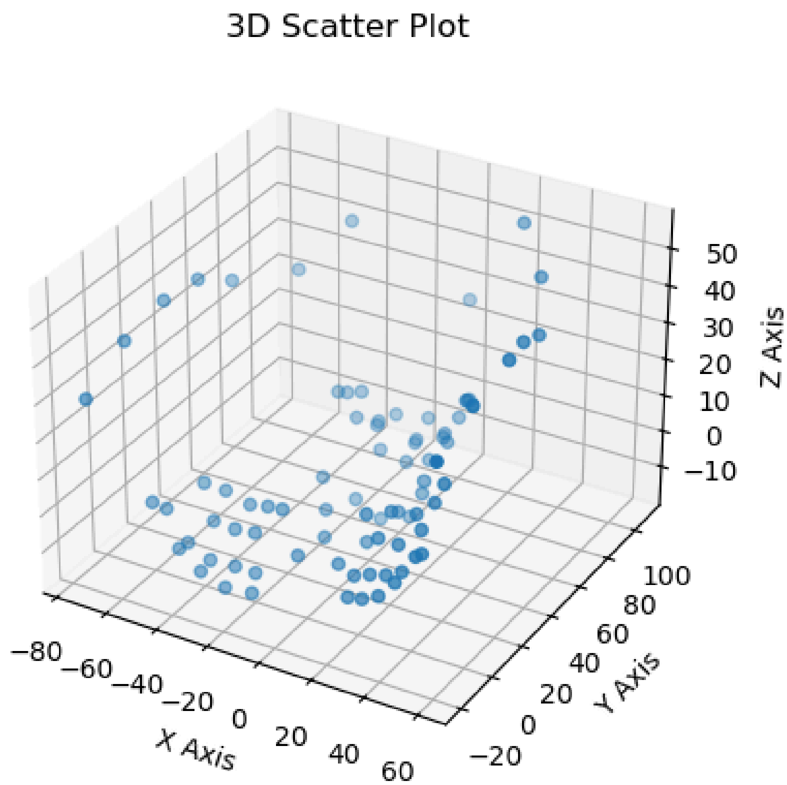
able to handle the possible non-linearities in the data due to its innate bias towards simpler models.

5. You must plot one sample of each data type – original, translated, rotated (1 each for x, y, and z). In total, you will have 5 figures/plots. Note: if you are unsure how to do this, look up a 3D scatter plot in python.

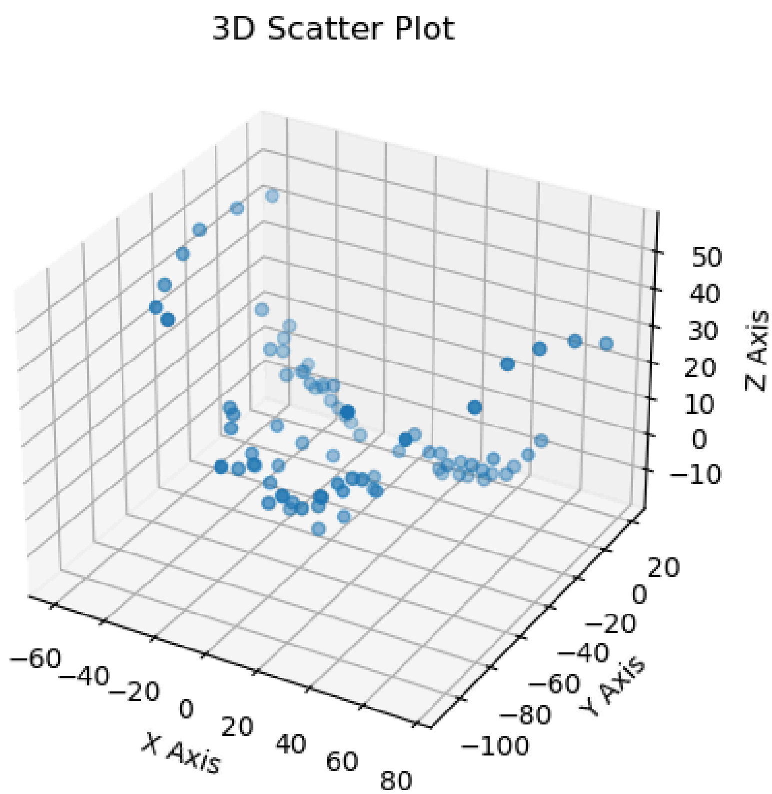
Original Data Sample



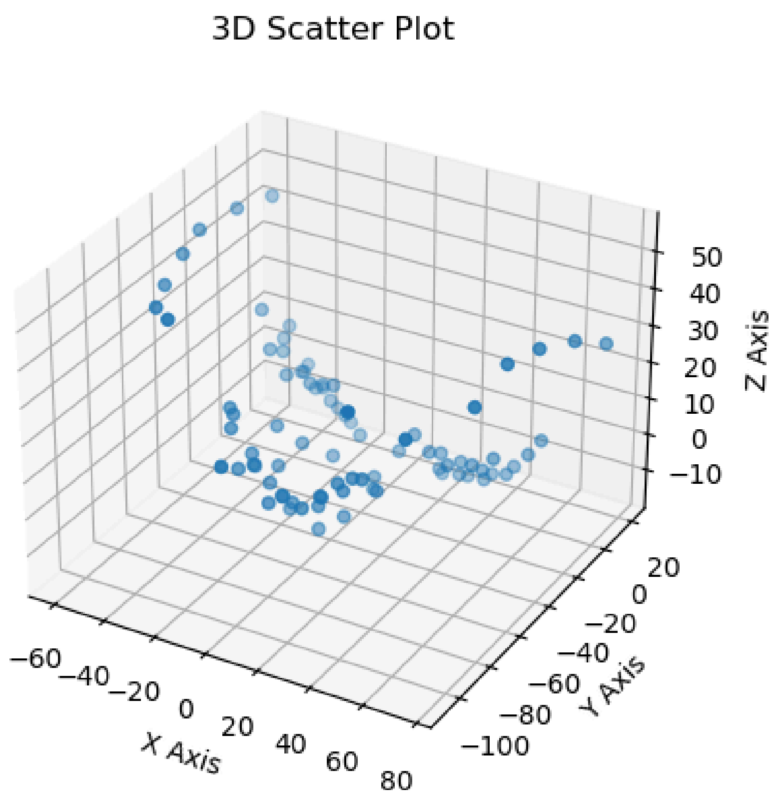
Translated Data Sample


3D Scatter Plot
X Axis
Y Axis
Z Axis
−80
−60
−40
−20
0
20
40
60
−20
0
20
40
60
80
100
−10
0
10
20
30
40
50

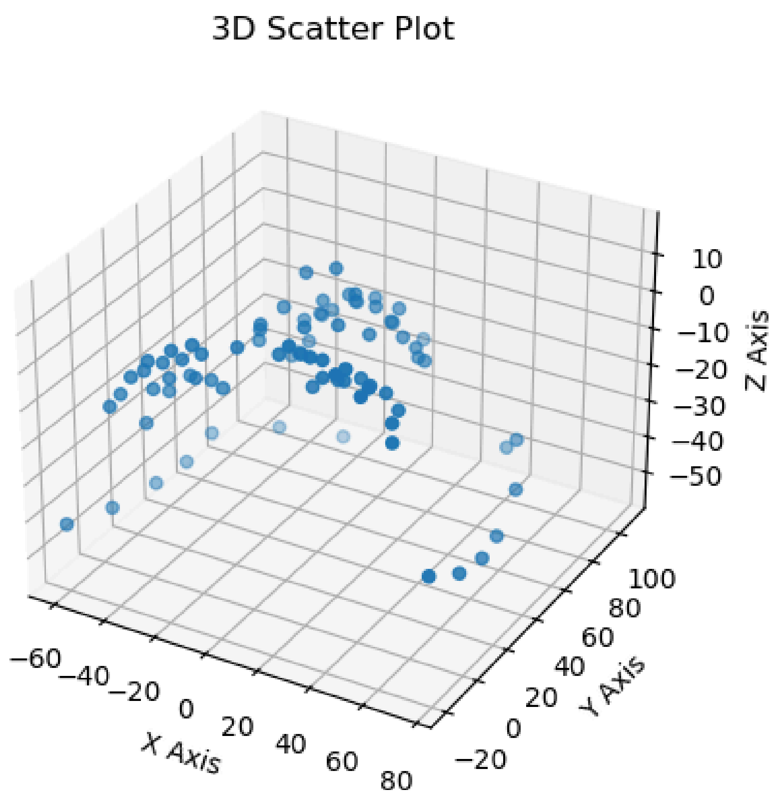
Rotated X Sample



Rotated Y Sample



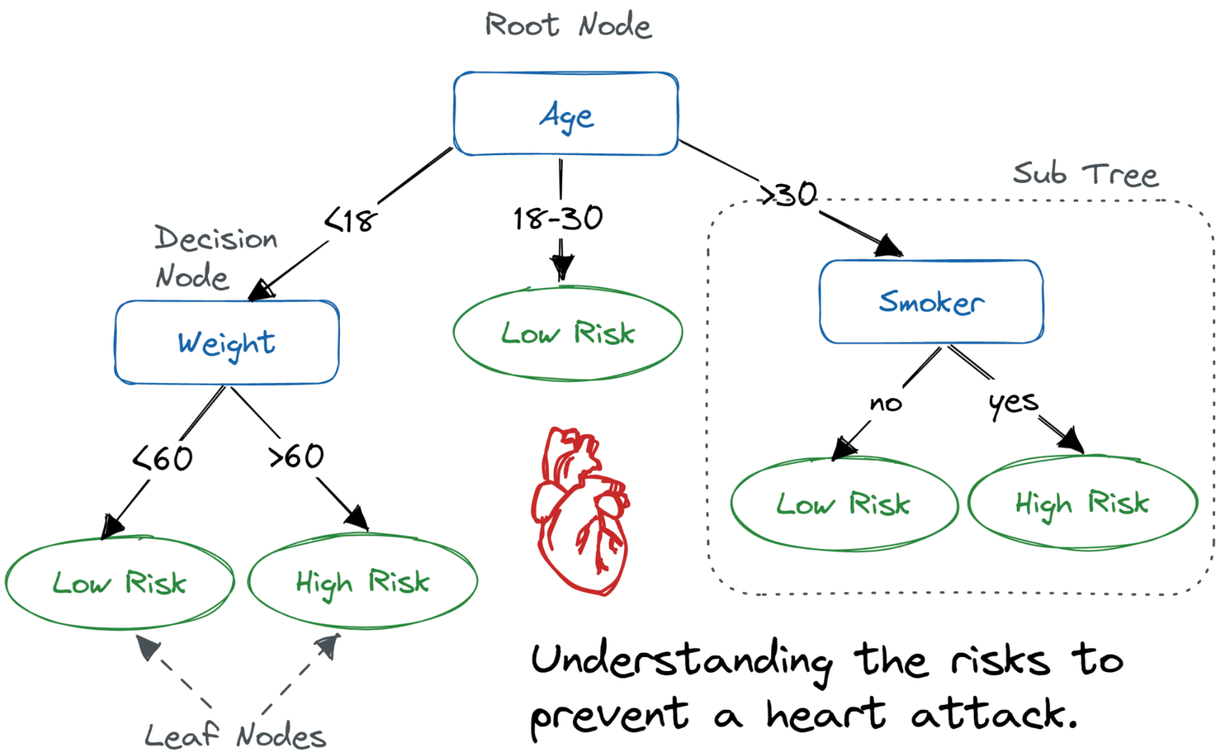
Rotated Z Sample



6. (CAP 5627 only) Describe, in your own words, what Random Forest, SVM, and Decision Tree classifiers are/how they work. You may need to look this information up. Your description should be approximately 100-150 words in length for each classifier.

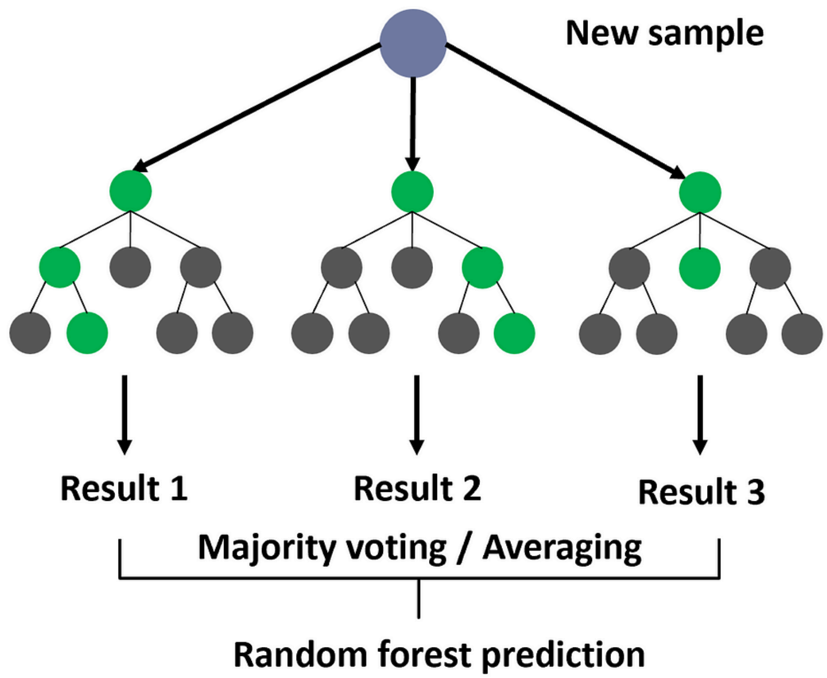
Decision Trees

Imagine a tree with branches (more like a flowchart) where you answer questions about data to arrive at each data node. A decision tree classifier works similarly. It splits the data based on features vectors (for example the coordinates in this case) at each step, asking a yes/no question. These splits keep happening until you reach a final category, like "happy" or "sad" or "angry" and so on. Decision trees are easy to understand but can be prone to overfitting if not carefully controlled.



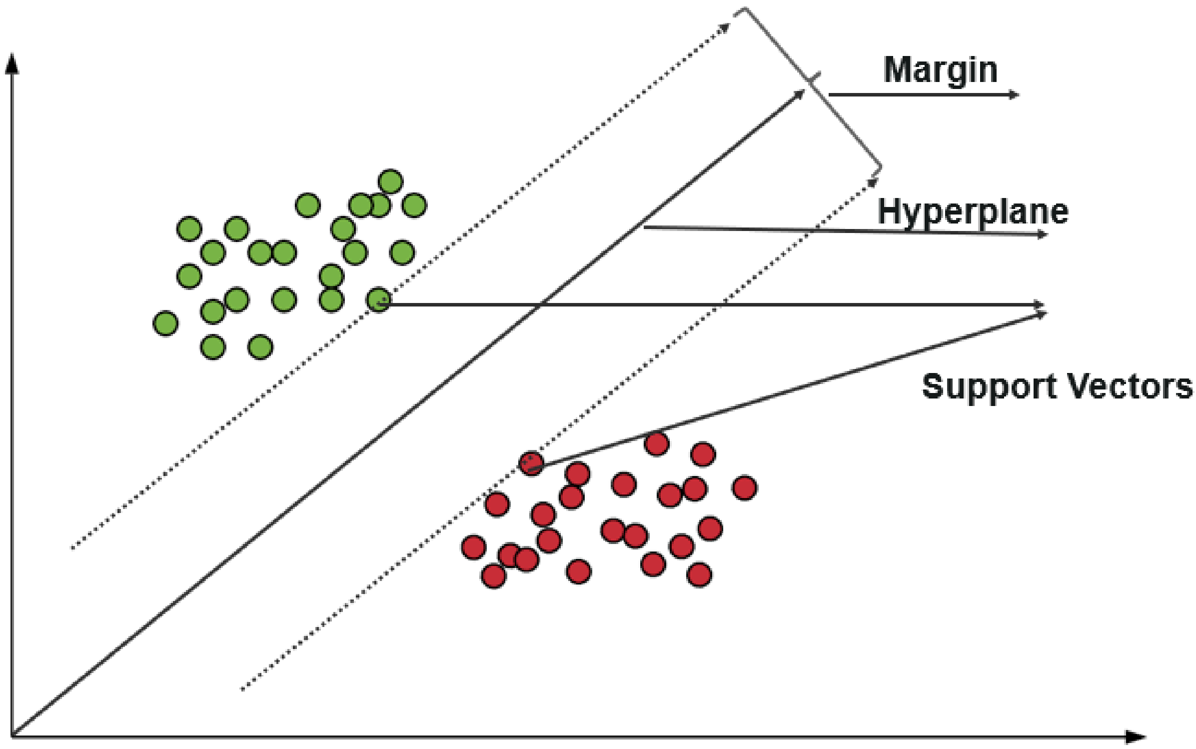
Random Forests

Random forests are like having a whole forest of decision trees! Each tree gets a random sample of the data and focuses on a random subset of features. Then, when a new data point comes along, all the trees vote on the category, and the most popular vote wins. This approach helps prevent overfitting and often leads to more accurate predictions than a single decision tree.



Support Vector Machines (SVMs)

Support Vector Machines imagine your data as points in a high-dimensional space. They find a hyperplane (a flat boundary in that space) that best separates the data points belonging to different categories. This hyperplane acts like a decision boundary - new data points are classified based on which side of the hyperplane they fall on. SVMs are powerful for high-dimensional data but can be computationally expensive to train for very large datasets.



Confusion matrix for top classifier:

SVM with Original Data						
True/Predicted	Angry	Disgust	Fear	Happy	Sad	Surprise
Angry	5388	2253	1355	901	3413	1003
Disgust	777	4106	1354	1045	286	870
Fear	873	1547	3598	1700	1157	1526
Happy	219	598	1456	5643	139	146
Sad	2541	1030	1175	482	4991	883
Surprise	326	637	1106	202	156	5520

SVM with Translated Data:						
True/Predicted	Angry	Disgust	Fear	Happy	Sad	Surprise
Angry	6473	2008	1203	1016	2722	936
Disgust	691	4993	1338	611	213	494
Fear	552	1534	4196	1569	638	1317
Happy	197	590	1462	6248	96	315
Sad	2005	673	1136	444	6432	895
Surprise	206	373	709	85	41	5991

SVM with RotatedX Data:						
True/Predicted	Angry	Disgust	Fear	Happy	Sad	Surprise
Angry	5523	2246	1413	926	3473	1033
Disgust	736	4154	1302	947	175	845
Fear	860	1538	3612	1666	1143	1536
Happy	186	601	1471	5720	151	129
Sad	2521	1039	1139	506	5054	913
Surprise	298	593	1107	208	146	5492

SVM with RotatedY Data:						
True/Predicted	Angry	Disgust	Fear	Happy	Sad	Surprise
Angry	5525	2232	1456	932	3483	1053
Disgust	734	4164	1265	845	155	832
Fear	871	1554	3662	1662	1140	1539
Happy	183	640	1468	5759	145	125
Sad	2516	1029	1114	564	5075	912
Surprise	295	552	1079	211	144	5487

SVM with RotatedZ Data:						
True/Predicted	Angry	Disgust	Fear	Happy	Sad	Surprise
Angry	5531	2227	1463	933	3493	1060
Disgust	734	4165	1245	807	160	833
Fear	881	1545	3700	1678	1112	1521
Happy	175	649	1461	5773	153	129
Sad	2510	1033	1117	574	5081	913
Surprise	293	552	1058	208	143	5492

All Classifiers (Section-wise):

Random Forest			
Metrics	Original	Data Translated	Rotated (Average)
Accuracy	0.3575	0.4991	0.3524
Precision	0.3472	0.4851	0.3434
Recall	0.3583	0.4995	0.352

Explanation:
Accuracy: The Random Forest classifier achieved an accuracy of 0.3575 on the original data, 0.4991 on the translated data, and an average accuracy of 0.3524 on the rotated data.
Precision: Precision refers to the ratio of correctly predicted positive observations to the total predicted positives. In this case, the Random Forest classifier achieved precision scores of 0.3472, 0.4851, and 0.3434 for the original, translated, and rotated data respectively.

Recall: Recall (also known as sensitivity) is the ratio of correctly predicted positive observations to the all observations in actual class. The Random Forest classifier achieved recall scores of 0.3583, 0.4995, and 0.3520 for the original, translated, and rotated data respectively.

SVM:			
Metrics	Original	Data Translated	Rotated (Average)
Accuracy	0.4842	0.5684	0.5164
Precision	0.5069	0.5905	0.5308
Recall	0.4845	0.5685	0.5148

Explanation:
Accuracy: The Support Vector Machine (SVM) classifier achieved an accuracy of 0.4842 on the original data, 0.5684 on the translated data, and an average accuracy of 0.5164 on the rotated data.
Precision: For precision, SVM obtained scores of 0.5069, 0.5905, and 0.5308 for the original, translated, and rotated data respectively.
Recall: SVM achieved recall scores of 0.4845, 0.5685, and 0.5148 for the original, translated, and rotated data respectively.

Decision Tree:			
Metric	Original	Data Translated	Rotated (Average)
Accuracy	0.272	0.358	0.2999
Precision	0.2662	0.3552	0.2946
Recall	0.2726	0.3586	0.3005

Explanation:
Accuracy: The Decision Tree classifier achieved an accuracy of 0.272 on the original data, 0.358 on the translated data, and an average accuracy of 0.2999 on the rotated data.
Precision: Precision scores for Decision Tree were 0.2662, 0.3552, and 0.2946 for the original, translated, and rotated data respectively.
Recall: Recall scores for Decision Tree were 0.2726, 0.3586, and 0.3005 for the original, translated, and rotated data respectively.