Gender Identification from Face Images

Tatiana Eng, Matt Kilmer, Orlando Marin

Topics of Discussion

- Team Members and Responsibilities
- Project Goals and Methods
- Framework
- Data and Features
- Experiments
- Results
- Conclusion

Team Members and Responsibilities

Team Members and Responsibilities

Orlando:

- Data cleaning and feature engineering
- Split the dataset into gender-balanced training and test sets based on unique person identifiers
- Implemented K-Nearest Neighbors (KNN) and Random Forest (extra model)
- Primary author of the corresponding report, with input from Tatiana and Matt

Tatiana:

- Implemented Artificial Neural Network (ANN) and Naive Bayes
- Created ROC curve and confusion matrix visualizations
- Created presentation slides, with input from Orlando and Matt

Matt:

- Implemented Decision Trees and Support Vector Machine (SVM)
- Created presentation slides, with input from Orlando and Tatiana

Project Goals and Methods

Overview

Goal:

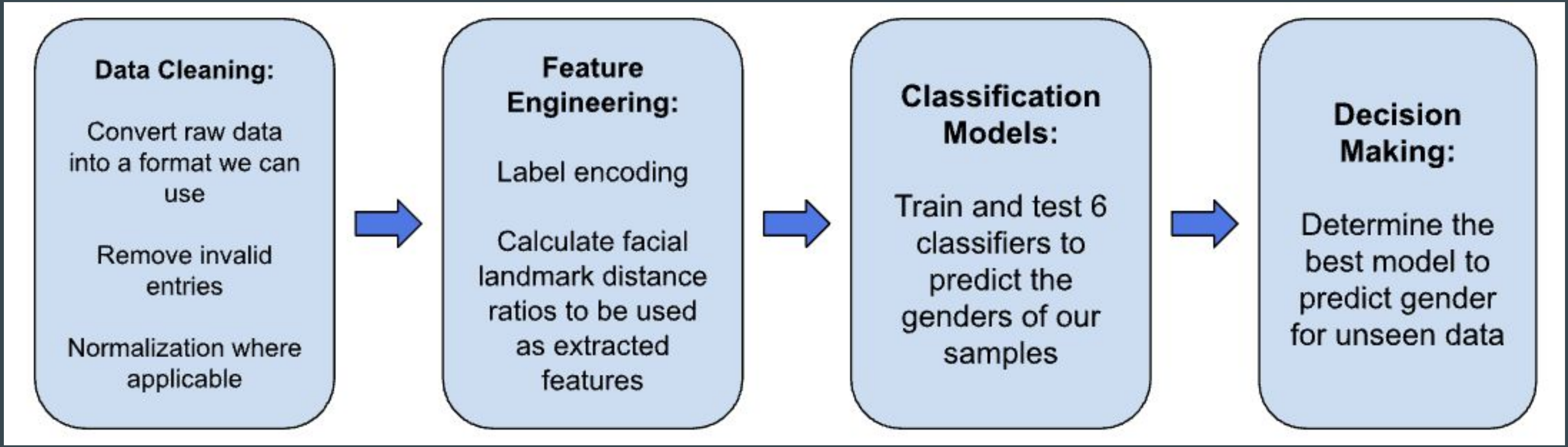
- Use facial landmark data and machine learning to identify a person's gender (male or female)
- Maximize models' accuracy, precision, and recall

Method:

- Extracted and calculated defining features
- Trained and tested 6 machine learning classifiers
  - K-Nearest Neighbor (KNN), Random Forest, Artificial Neural Network (ANN), Naive Bayes, Decision Tree, Support Vector Machine (SVM)
  - Divided the data into male and female, then did an 80% / 20% train-test split for each gender's data
    - For each gender, 80% of the unique IDs were used for the training and 20% for testing
    - Ensures that no individual appears in both the training and testing set (no data leakage)
- Evaluated model performance based on confusion matrices and AUC (area under curve)

Framework

Framework



Data Cleaning:
Convert raw data into a format we can use
Remove invalid entries
Normalization where applicable
Feature Engineering:
Label encoding
Calculate facial landmark distance ratios to be used as extracted features
Classification Models:
Train and test 6 classifiers to predict the genders of our samples
Decision Making:
Determine the best model to predict gender for unseen data

Data and Features

Data and Features

- Dataset: AR Face Database
  - 509 facial images that each contain 22 markup points (x, y coordinates)
  - Images are of 136 subjects (76 males, 60 females)
- Features (Extracted):
  - Eye length ratio
  - Eye distance ratio
  - Nose ratio
  - Lip size ratio
  - Lip length ratio
  - Eyebrow length ratio
  - Aggressive ratio
- Target:
  - gender_label

Data Cleaning

- Removed rows with invalid entries
  - 1 row with expression "05_a"
- Parsed gender and metadata from file names
  - Label encoded gender to create the "gender_label" column
  - 0 represented female, and 1 represented male
- Standardized data for KNN, ANN, and SVM
  - Ensures that each feature contributes equally to distance-based calculations

Feature Engineering

- Created functions to calculate each of the 7 extracted features using Euclidean distance

Experiments

K-Nearest Neighbor (KNN)

Parameter tuning:

- Tested values of k between 1 and 50, inclusive

Parameter that yielded the highest accuracy:

- k = 18

Results

- Accuracy: 71.82%
- Precision: 73.24%
- Recall: 81.25%

Random Forest

How Random Forest works:

- Builds many decision trees using random subsets of the data and features
- Makes predictions by combining the results of all trees, which typically improves accuracy and reduces overfitting

Parameter tuning:

- Number of trees (n_estimators)
- Maximum depth of each tree (including no max. depth)

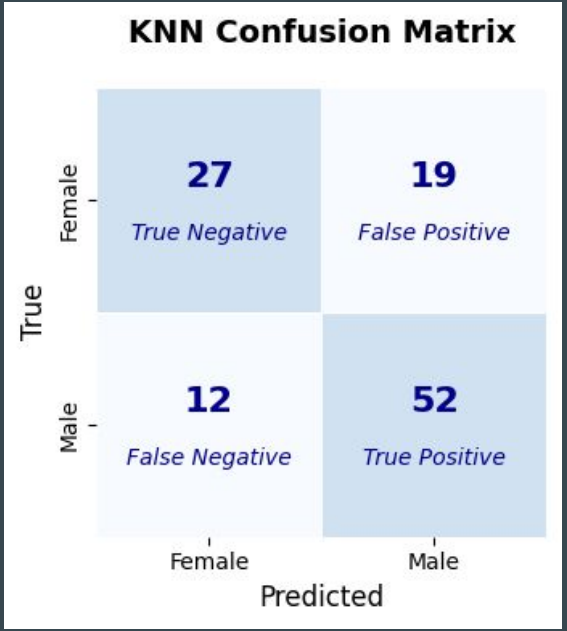
Parameters that yielded the highest accuracy:

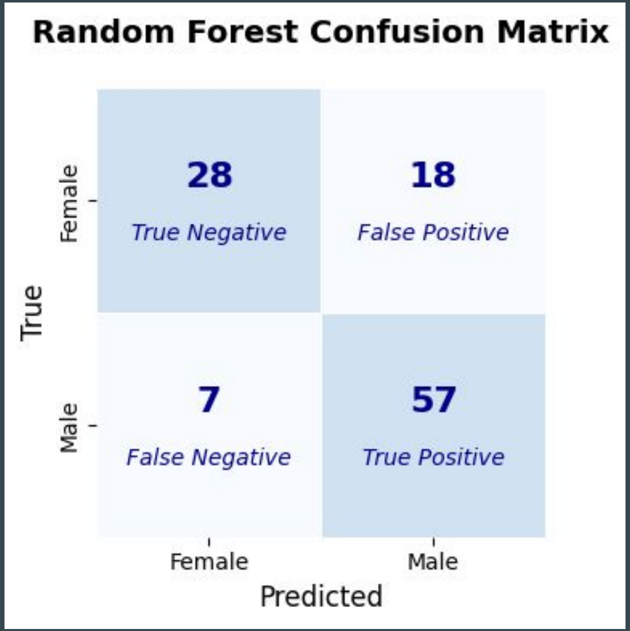
- Number of trees: 30
- Depth of each tree: 15 layers

Results

- Accuracy: 77.27%
- Precision: 76%
- Recall: 89.06%

K-Nearest Neighbor and Random Forest Confusion Matrices





KNN made 79 correct predictions. Random Forest made 85 correct predictions. 110 total predictions per model.

Artificial Neural Network (ANN)

Parameter tuning:

- Tested different learning rates, tolerances, numbers of hidden layers, and number of neurons in each hidden layer

Parameters that yielded the highest accuracy:

- Learning rate: 0.01
- Tolerance: 0.01
- 1 hidden layer consisting of 100 neurons

Results:

- Accuracy: 76.36%
- Precision: 75.68%
- Recall: 87.50%

Naive Bayes

Gaussian Naive Bayes

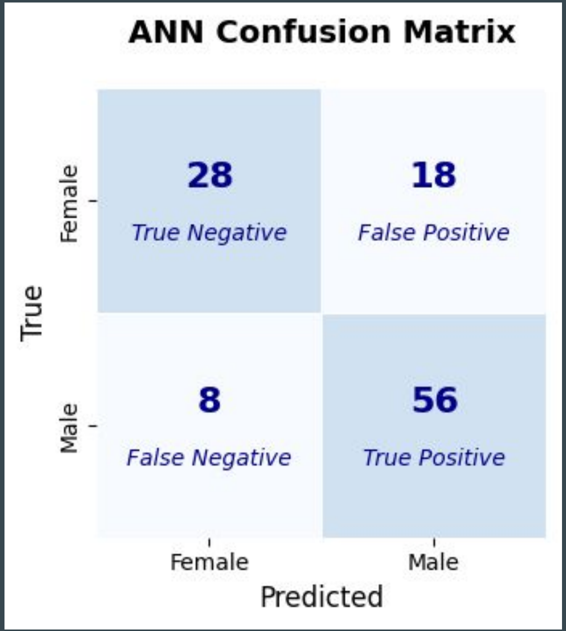
- Best for continuous data
- We assume the data follows a normal distribution

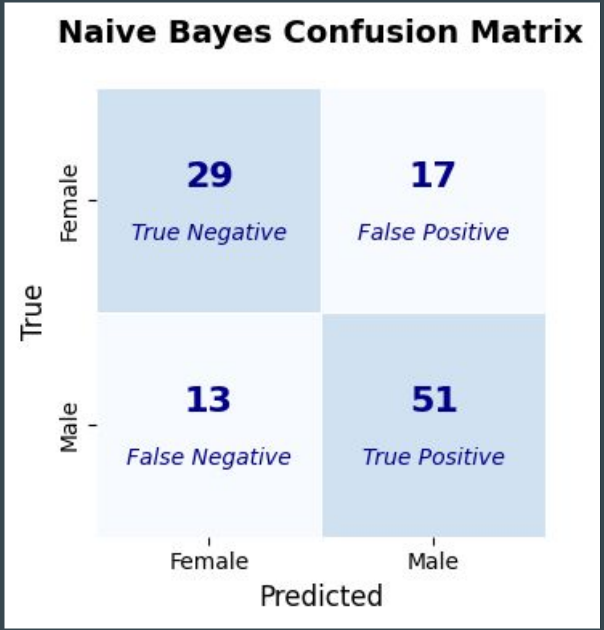
Gaussian Naive Bayes is a probability model that primarily relies on mathematical formulas. As a result, parametric tuning is not applicable.

Results:

- Accuracy: 72.73%
- Precision: 75%
- Recall: 79.69%

Artificial Neural Network and Naive Bayes Confusion Matrices





ANN made 84 correct predictions. Naive Bayes made 80 correct predictions. 110 total predictions per model.

Decision Tree Classifier

Parameter tuning:

- Maximum depth of each tree.
- Minimum number of samples needed to split a node.
- Minimum number of samples needed in a leaf.

Parameters that yielded best accuracy:

- max_depth = 8
- min_samples _split = 2
- min_samples _leaf = 5

Results:

- Accuracy: 63.64%
- Precision: 67.65%
- Recall: 71.88%

Support Vector Machine (SVM)

Parameter tuning:

- Type of kernel used to separate data.
- C-value, which controls the give-and-take between simplicity and accuracy.

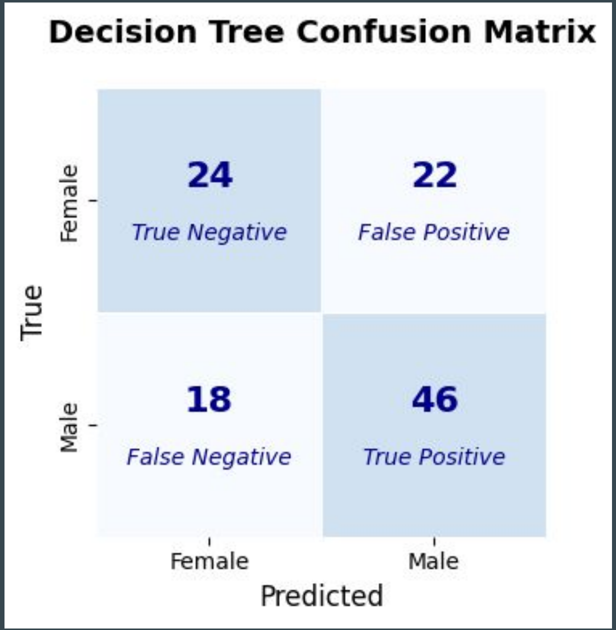
Parameters that yielded best accuracy:

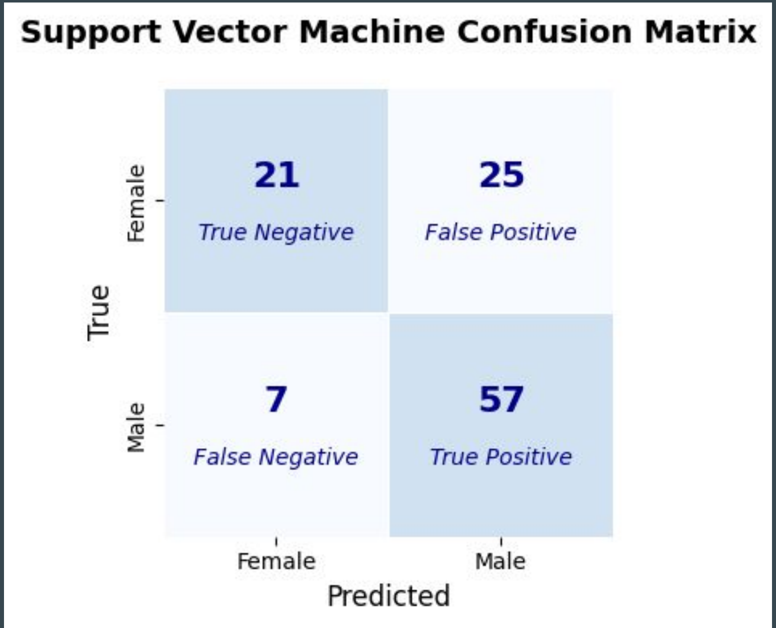
- Kernel: RBF (Radial Basis Function)
- C = 1

Results:

- Accuracy: 70.91%
- Precision: 69.51%
- Recall: 89.06%

Decision Tree and Support Vector Machine Confusion Matrices





Decision Tree made 70 correct predictions. Support Vector Machine made 78 correct predictions. 110 total predictions per model.

Results

Results: Accuracy, Precision, and Recall Table

Random Forest produced the highest accuracy and precision and tied with Support Vector Machine for the highest recall.

Model	Accuracy	Precision	Recall
KNN	71.82%	73.24%	81.25%
Random Forest	77.27%	76%	89.06%
ANN	76.36%	75.68%	87.50%
Naive Bayes	72.73%	75%	79.69%
Decision Tree	63.64%	67.65%	71.88%
SVM	70.91%	69.51%	89.06%

Results: Associated Confusion Matrices

KNN

TN	27	19	FP
FN	12	52	TP

ANN

TN	28	18	FP
FN	8	56	TP

Decision Tree

TN	24	22	FP
FN	18	46	TP

Random Forest

TN	28	18	FP
FN	7	57	TP

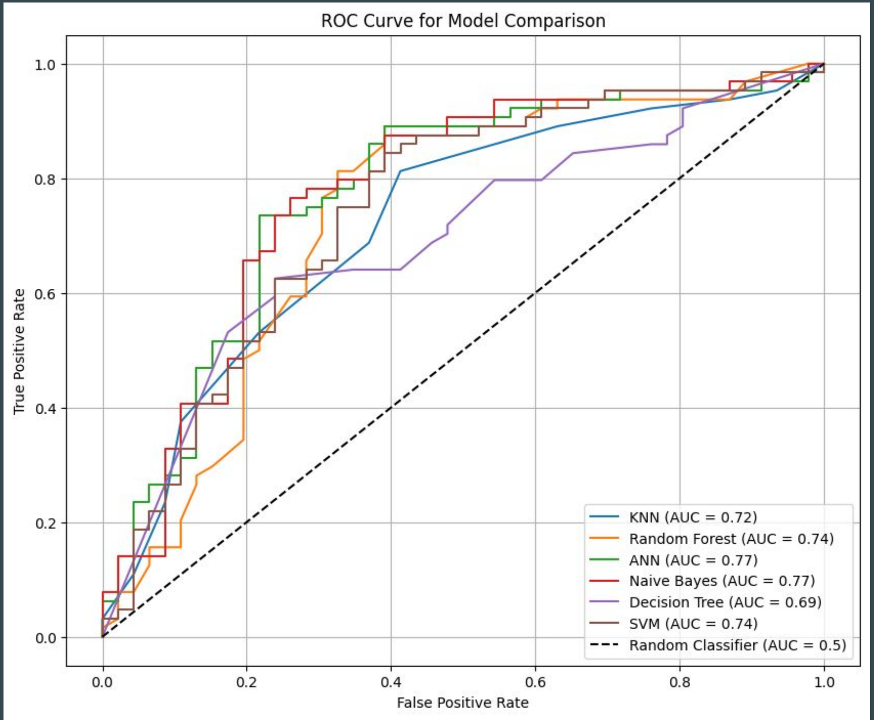
Naive Bayes

TN	29	17	FP
FN	13	51	TP

SVM

TN	21	25	FP
FN	7	57	TP

Receiver Operating Curve (ROC) and Area Under the Curve (AUC)



An ROC curve plots the true positive rate (TPR) against the false positive rate (FPR) at varying thresholds.

The area under the curve (AUC) indicates the model's performance across all thresholds.

- AUC = 1 → perfect model with a 100% TPR
- AUC = 0.5 → model is essentially making random guesses
- AUC < 0.5 → worse than random guessing

ANN and Naive Bayes have the highest AUC at 0.77.

Conclusion

Conclusion

- Key Takeaway:
  The 7 engineered facial features can effectively be used to perform gender classification.
- Random Forest had the highest accuracy (77.27%) but ANN and Naive Bayes had the highest AUC value (0.77).
  - This means that Random Forest made the most correct predictions compared to the other models, while ANN and Naive Bayes have the best ability to distinguish between classes.
- Decision Tree performed the worst overall and had the lowest accuracy (63.64%), precision (67.65%), and recall (71.88%).

Limitations

- Class Imbalance:
  - Out of 136 total subjects, there were 76 male vs. 60 female subjects.
  - 301 image samples for men vs. 208 image samples for women.
  - Future work: Use a balanced dataset to reduce bias and overrepresentation.
- 22 landmarks per image: limits the precision of facial detail that could be captured.
  - Future work: Use additional extracted features to better capture differences between male and female facial structure and lead to potentially better model accuracy.
- Absence of vital categorical data (age group, skin tone, hair length, etc.).
  - Future work: Include additional categorical information about each image to provide models with more context during training.

Thank you!

Appendix

References

- AR Face Database
- https://www.geeksforgeeks.org/machine-learning/random-forest-algorithm-in-machine-learning/
- https://www.geeksforgeeks.org/machine-learning/naive-bayes-classifiers/