Final Project: Multimedia Classification System

Universidad Carlos III de Madrid, Multimedia, Year 2019-2020

Authors
Naman Agarwal – Justin Spar

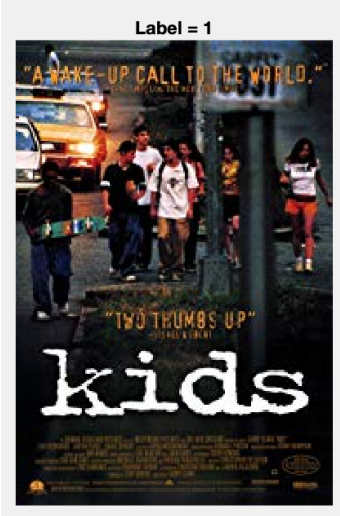
Q1.

HSV separates out the colors from the saturation and intensity, which is usually more valuable than knowing the amount of red/green/blue in a pixel. For example, shadows can drastically change RGB values but the HSV separates brightness from the color itself.

Q2.

The remaining channel is saturation, which can be incredibly useful since it separates grayscale from very deeply colored images. One example hypothesis in the context of movie posters would be that drama films are more likely to have unsaturated posters, since grayscale often makes images more “dramatic.”

Q3.



Brightness (V): 0.25 **Brightness (V)**: 0.74

For the darker poster, the average value of HSV (V) is lower than the average value of V for the lighter poster. That shows that the light emitted from an average pixel of the lighter poster is higher than the amount of light emitted from an average pixel of the darker poster.

Q4.

It would not be as useful since larger images would be skewed to have more edges. The same image at a lower resolution would have fewer edges. This issue can be resolved by obtaining the edge density, or the ratio of edge to non-edge pixels.

Q5.

An image in the input data represents 268*182*3 = 146328 values. This gives, after reducing this to 3 values, a savings ratio of 1 - 3/146328 = 0.99998.

Q6.

Otsu's method is used to perform image thresholding; it returns a single intensity threshold that separate pixels into two classes, foreground and background. In the case of movie posters, Otsu's method can be used to separate foreground items such as the title from the poster background.

Q7.

The training feature matrix will have 4800 elements. This is because we are now extracting 5 features (2 additional with the text extraction: number of words and mean word length in the description) from each of the 960 images.

Q8.

5.5 MB * 10^6 = 5500000 B (text)
5500000 + (24 * 268 * 182 * 960) = 1,129,299,040 B (images + text)

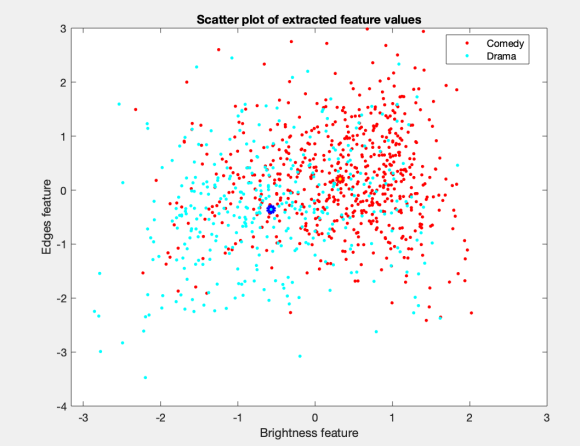
8 B * 5 features * 960 movies = 38400 B (feature matrix)

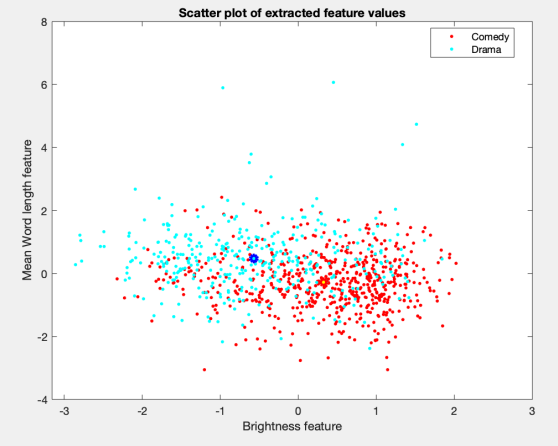
38400 / 1,129,299,040 = 0.00003400339382
1-0.00003400339382 = 0.99997 (savings ratio)
Expressed as a percentage: 99.997%

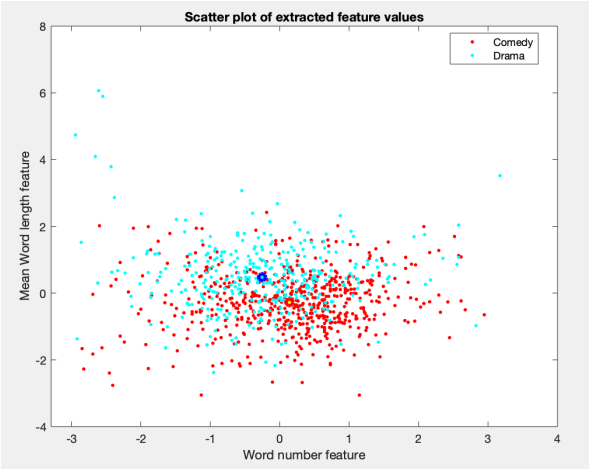
Q9.

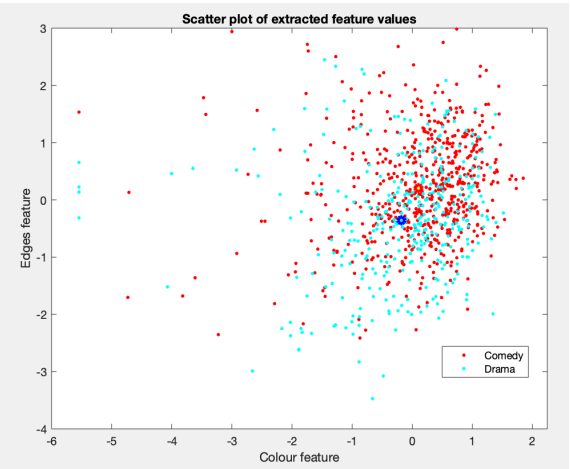
Filler words like "the", "of", and "to" will occur most frequently within the words vector and in the overall corpus of documents, but won't be of any use while trying to distinguish the genre of a given movie. The words that would be most useful while performing classification would be "maid", "Romance", and "wedding" since they indicate semantic meaning. (This is why we would use a weighting factor like tf-idf before performing classification.)

Q10.









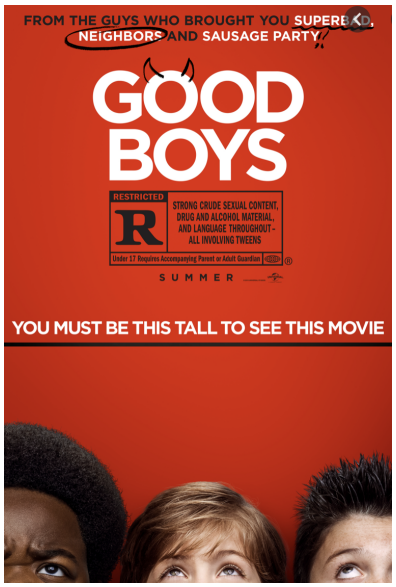
Brightness is the most useful feature for genre discrimination, which makes sense since lighthearted and happy comedy movies will typically have brighter posters than darker, more somber dramas. There are also combinations with brightness that can be used for genre discrimation, like brightness and mean word length and brightness and number of edges. The brightness-edge combination appears to perform the best based on the scatter plots.

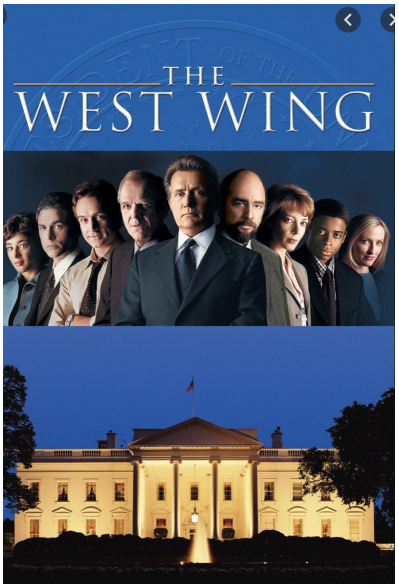
Q11.

One useful feature we could additionally extract could be saturation. More serious movies like dramas might have less saturated posters, since dramas tend to have more black and white stark contrast and less “pop” in their colors than comedy movies.

Another interesting feature to work with for classification could be hue (it’s already being extracted as part of HSV but we aren’t actually using it for the discrimination task). While over the aggregate dataset, there’s a risk that the complete range of hue values will be represented with both comedy and drama movies, there’s psychology around color theory that indicates how certain colors make people feel. For example, red evokes more passion while blue evokes a

more calm and serious feeling. It's possible that movie directors or their marketing agencies take this color theory into account while designing posters. For example, the below comedy poster is primarily red and the drama is primarily blue. It's worth noting that we would need to condense the discrete variable of hue into a categorical variable while cleaning the data and preparing to build a classifier (and additionally, we'd need to use a one-hot encoder or similar in order to build a Bayesian classifier with categorical data).





Q12.

Since a feature can't have covariance with itself, the resulting matrix will be a 1x1 mean value. Features can't have covariances with themselves since those values are based on comparing the mean of a feature with means of the other features.

Q13.

In order for the decision boundary to be equidistant from the means of each class, each class must have the same variance. This is because the boundary falls where the two Gaussian curves meet, and if one curve has a higher variance it'd be wider than the other class's curve. If one curve is wider than the other, than the decision boundary will be further away from the mean of the wider curve than the narrower one.

Q14.

Changing the normalization parameters now would be changing the model -- we want the model to classify based on the normalized training data. If it was renormalized to the test dataset, then the classification results will be inaccurate since it will be splitting the data along a different threshold than before. (Similar example of this -- logistic regression models become less

accurate when upsampling a minority class to balance the data, since the model is then trained expecting a balanced split of future data.)

Q15.

We can't test on the training set since the model is attuned to the training set, so running the model against its training data would result in a perfect score. Splitting the data is necessary in order to evaluate the model accurately, since the model's performance is measured by how it performs against data it hasn't "seen" yet.

Q16.

An excessively small training set may lead to an inaccurate model -- without a lot of representative data, the model will be tuned based on the smaller distribution of data that it has seen. If there's a small training set with a high ratio of outliers, for example, then the resulting model will be trained to classify future data with the assumption that future data will have the same shape, even though it won't. An excessively small test set would make it difficult to evaluate the model's performance, since we won't be able to accurately converge to an accuracy/F1 score with just a few data points.

Q17.

The Gaussian predictor is essentially a Naive Bayes classifier that has no prior estimates (it assumes the data comes from a normal distribution). First, the model designates the `p_pos` as the normal probability density function of a movie being scored as a 1 using the mean and covariance matrix of positive training data from the model. Then, it saves the `p_neg` as the probability density function of negative samples. It subtracts the logs of the two pdf functions in order to set the limits (making sure the predicted scores fall within the same range). Finally, if the predicted score is above 0, it sets the predicted label to 1, and if not, it sets the predicted label to -1.

Q18.

P_D (detection) = 0.63
P_{FA} (false alarm) = 0.20

The probability of detection represents the probability that a positive sample will be correctly identified, while the probability of false alarm is the probability that a negative sample will be identified as positive (producing a false alarm). In our case, since a positive sample is a comedy, the probability of detection represents the probability that a comedy will be correctly identified, while the probability of false alarm is the probability that a drama will be mistakenly identified as a comedy.

Q19.

AUC (I) = 0.79
AUC (T) = 0.69
AUC(I+T) = 0.81

Based on these numbers, the third model offers the best performance. This makes sense, since it is classifying based on both visual features and text features, rather than either feature type by itself. It also makes sense that the first model performs better than the second, as the visual features are more useful than the chosen text features for comedy/drama classification.

Q20.

An AUC of 1 would imply that the model is perfect, which is impossible, while an AUC of 0.5 means that your model is classifying completely randomly.