## Milestone 3: Traditional statistical and machine learning methods, due Wednesday, April 19, 2017

Think about how you would address the genre prediction problem with traditional statistical or machine learning methods. This includes everything you learned about modeling in this course before the deep learning part. Implement your ideas and compare different classifiers. Report your results and discuss what challenges you faced and how you overcame them. What works and what does not? If there are parts that do not work as expected, make sure to discuss briefly what you think is the cause and how you would address this if you would have more time and resources.

You do not necessarily need to use the movie posters for this step, but even without a background in computer vision, there are very simple features you can extract from the posters to help guide a traditional machine learning model. Think about the PCA lecture for example, or how to use clustering to extract color information. In addition to considering the movie posters it would be worthwhile to have a look at the metadata that IMDb provides.

You could use Spark and the ML library (https://spark.apache.org/docs/latest/ml-features.html#word2vec) to build your model features from the data. This may be especially beneficial if you use additional data, e.g., in text form.

You also need to think about how you are going to evaluate your classifier. Which metrics or scores will you report to show how good the performance is?

The notebook to submit this week should at least include:

- Detailed description and implementation of two different models
- Description of your performance metrics
- Careful performance evaluations for both models
- Visualizations of the metrics for performance evaluation
- Discussion of the differences between the models, their strengths, weaknesses, etc.
- Discussion of the performances you achieved, and how you might be able to improve them in the future

## Description of Models

### Baseline: Logistic Regression

We selected a logistic regression to act as a baseline model. Multiple logistic regression is a simple model and especially appropriate when we are predicting a binary response. It also allows us to see the weights for each feature and get a sense of which predictors are important. We tuned the regularization constant and regularization method (L1 vs L2) using three-fold cross validation and sklearn's OneVsRestClassifier, which allows us to tune the models for all genres simultaneously (full multilabel approach).

### SVM

We selected an SVM as these models generally produce very good classifications and they are robust to noise and less prone to overfitting. The major disadvantage to the SVM, which we ran into, is that it's computationally expensive. We tried to speed up tuning time by running PCA and retaining the principle components that accounted for 90% of the variance. We also tried randomly sampling a proportion of the training data with which to tune on. We ran the SVM using five-fold cross validdation and performed grid search with the parameters kernel (rbf, linear), gamma, and cost.

### Random Forest

We selected to use a Random Forest as an alternative to the SVM, both for its general classification accuracy and relative training efficiency. Random forests are generally good candidates for classification tasks due to their ability to combine multiple predictions into an ensemble in order to reduce model variance. However, random forests are also susceptible to overfitting, particularly in instances where the data are sparse. In this dataset, we retained several numerical features from the original TMDB/IMDB datasets, and constructed bag-of-words representations of the titles and overviews, retaining the top 100 most frequent words for each feature. While still sparse, we expected that retaining only the most frequent words would reduce the overall sparsity and allow the RF to perform relatively well. In order to reduce overfitting, we tuned the Random Forest using five-fold cross validation and performed grid search using three values each for max_features, max_depth, and num_estimators.

In addition to the multiclass version of this problem, a multilabel RF was also trained and scored below.

## Logistic Regression: Most important predictors

One benefit of using a regularized logistic regression classifier is that it is easy to extract weights and understand which features are most important to the model.

Below are the top 5 most (positively correlated) predictors for each genre:

**Action** ['popularity' 'languages_serbo' 'languages_japanese' 'languages_russian' 'languages_greek']

**Adventure** ['green_pixel' 'countries_japan' 'countries_brazil' 'languages_tagalog' 'languages_english']

**Animation** ['countries_japan' 'cast_melblanc' 'green_pixel' 'countries_southkorea' 'languages_thai']

**Comedy** ['overview_comedy' 'red_pixel' 'director_davefleischer' 'color info_path' 'cast_hankbell']

**Crime** ['languages_japanese' 'overview_murder' 'languages_serbo' 'languages_hindi' 'budget']

**Documentary** ['rating' 'overview_documentary' 'color info_color' 'writer_nan' 'languages_english']

**Drama** ['overview_drama' 'runtime' 'votes' 'rating' 'writer_howardj']

**Family** ['green_pixel' 'overview_children' 'director_chuckjones' 'cast_jeffreysayre' 'vote_count']

**Fantasy** ['languages_tagalog' 'countries_japan' 'cast_shea' 'countries_southkorea' 'vote_average']

**History** ['rating' 'red_pixel' 'cast_nan' 'overview_high' 'countries_czechoslovakia']

**Horror** ['color info_color' 'languages_turkish' 'vote_count' 'countries_mexico' 'writer_mickgarris']

**Mystery** ['blue_pixel' 'overview_murder' 'runtime' 'languages_spanish' 'countries_croatia']

**Romance** ['overview_love' 'red_pixel' 'cast_rayjones' 'runtime' 'countries_india']

**Science Fiction** ['color info_color' 'countries_japan' 'countries_mexico' 'cast_robertj' 'overview_world']

**Thriller** ['popularity' 'cast_hermanhack' 'languages_turkish' 'color info_color' 'vote_average']

**War** ['overview_war' 'countries_yugoslavia' 'languages_korean' 'languages_german' 'languages_tamil']

**Western** ['countries_usa' 'green_pixel' 'cast_artdillard' 'director_nan' 'color info_technicolor']

## Performance Metrics

### F1 Score

F1 measures the balance between precision (exactness) and recall (sensitivity) scores and is calculated as:

$$\frac{2 * (precision * recall)}{precision + recall}$$

For this task, we are concerned by the imbalance among the genres in the dataset. We determined that one way to account for this problem during training was to use the F-score for model tuning; by balancing precision and recall, the model with the "best performance" will not be one that selectively performs well only on the dominant class.

### Hamming Loss

This metric measures model accuracy for multi-label classification problems. Although we implemented multi-class models above, we constructed "multi-label" final outputs by combining each genre classifier's predicted output into a multi-label Y. We then computed the hamming loss on this final output in order to compare the three strategies (Logistic Regression, RF, and SVM).

Hamming loss is given by the following formula:

$$\frac{1}{|D|} \sum_{i=1}^{|D|} \frac{xor(x_i, y_i)}{|L|}$$

where $|D|$ is the number of observations, $|L|$ is the number of labels, $y_i$ are the actual labels, and $x_i$ are the predicted labels.

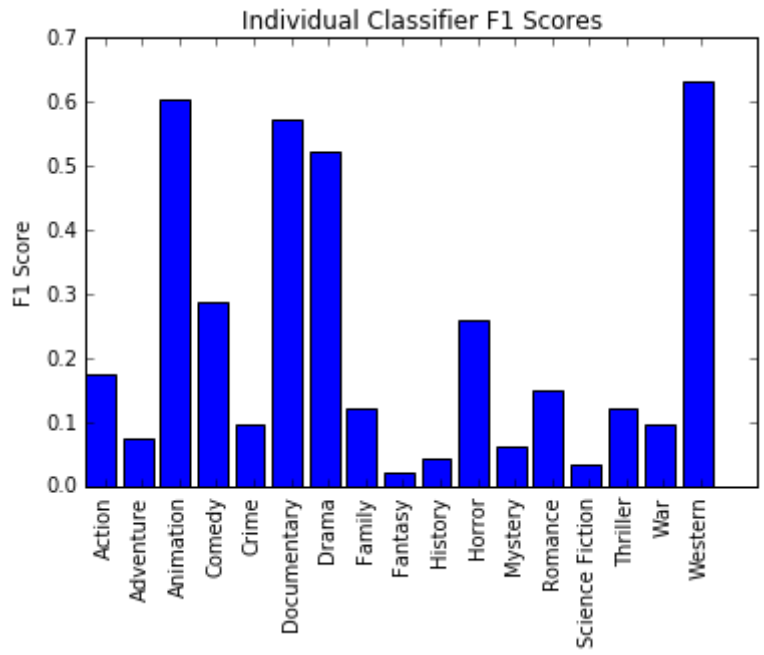
Source: (https://www.kaggle.com/wiki/HammingLoss (https://www.kaggle.com/wiki/HammingLoss))

## Performance Evaluation

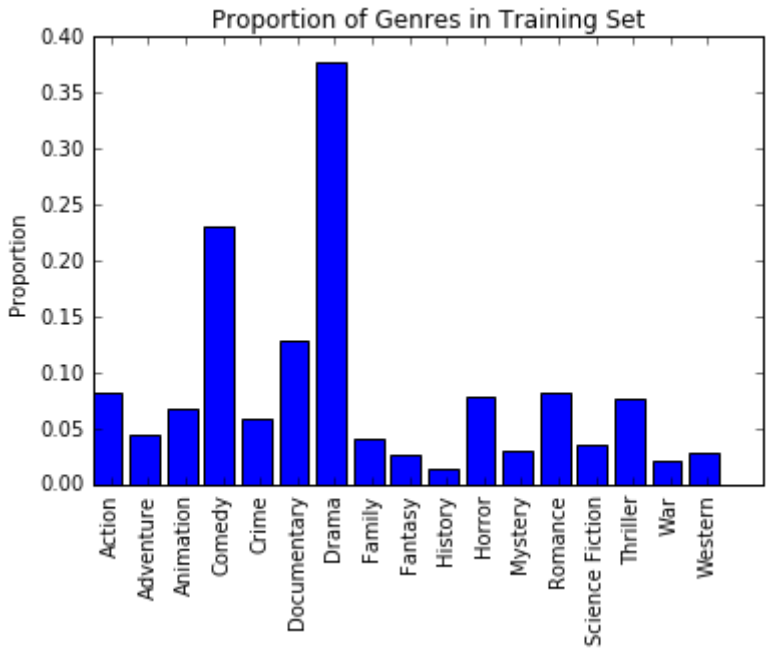
The following scores are reported for the test set:

| Model | Hamming Loss |
|---|---|
| Logistic Regression | 0.21 |
| Multiclass Random Forest | 0.073 |
| Multilabel Random Forest | 0.075 |
| SVM | 0.23 |

For the most successful classifier, the Random Forest, we extracted the F1 scores for each multiclass model as a measurement of the predictive quality for individual genres.




We observe that the Multiclass Random Forest still underperforms on a number of smaller genres. However, a number of unpopular genres have very high F1 scores -- Westerns, for example, comprise less than 5% of the dataset but have the highest F1 score. For comparison, the distribution of genres in the dataset is shown below.




## Discussion of Models and opportunities for improvement

The SVM model performed the worst in terms of Hamming loss of the three models. We attribute this to the fact that the SVM was too computationally expensive to tune with a finer grid-search. Further, while sampling did speed up the computation time, it decreased the f1 score of particularly the more obscure genres that are not common in our dataset. In terms of model improvement there are two clear paths forward. Firstly, the results would almost certainly improve if we used a wider array of values for the hyperparameters in tuning. Secondly, sampling could be preformed in a stratified manner, thus decreasing the computation time while maintaining enough of each genre to make an accurate classifier.

The logistic regression acted mostly as a baseline, but with more time we could tune this model further as well. The class weights in particular could benefit from further tuning.

While the Random Forest did not display preferential treatment for only the most frequent genres in the test set, it continued to underperform on certain genres. We also used the multilabel variant of the RF model, which had very similar, but marginally worse, performance. For both the multiclass and multilabel cases, we found that sklearn's implementation essentially treats each genre as independent of the others (via multilabel binarization in the multi-label case). We suspect the assumption of independence might have an effect on the model's overall hamming loss. We would like to be able to leverage information about genres that typically occur together. For example, if a genre is labeled a Action, there is some likelihood that it is also labeled as an Adventure movie. If these genres are treated independently, then identifying the movie as Action has no bearing on whether it is also labeled Adventure. This ultimately effects the hamming loss, since the metric increases with the correct identification of all labels. In order to account for relationships among genres, we could transform the most common genre pairings into their own genre labels in the response variable.

In [ ]: