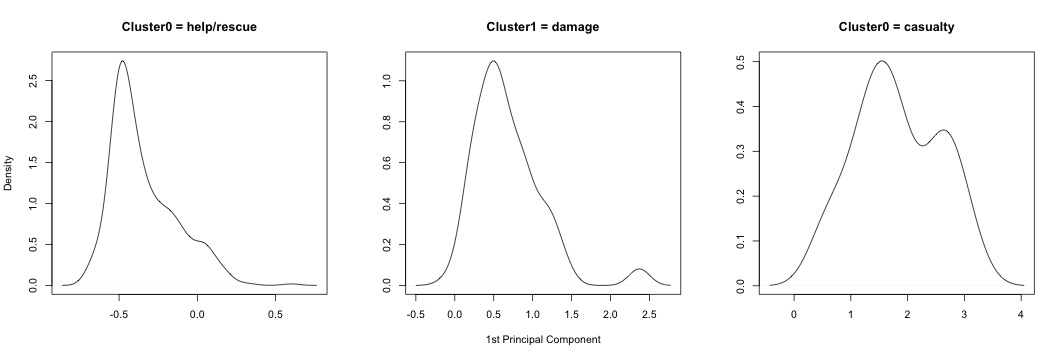
Tweet Analysis:

The purpose of analyzing tweets on a natural disaster is to find the context of the conversation during and after the disaster has taken place. We applied unsupervised techniques on 596 tweets about Ellicott City Flash Flood to first learn about the context to categorize them and then train a model to classify unseen tweets. This process was followed by two steps – i) cluster the tweets to identify categories and ii) train a model to classify unseen tweets to fit one of the categories.

**Clustering:** In the first step we used, Latent Semantic Analysis to find tweet similarity. We used tf-idf weighing scheme to vectorize the tweets. We accepted terms that appeared in at least two tweets and ignored the terms that appeared in more that half of the tweets. Next, we applied Singular Value Decomposition for dimensionality reduction.

We took the top k =150 singular values, which explained 90% of the variance. We computed a tweet-tweet cosine similarity matrix from the right - singular matrix (V). After that, we performed kmeans clustering on the 1- cosine similarity matrix to group the tweets based on their similarity with other tweets. We identified three categories by analyzing terms with the highest tf-idf in the clusters. The three categories are “help/rescue”, “damage/aftermath” and “casualty”. Following are the density plots of the first principal components of the clusters. We



**Classification:** After labeling all the tweets with thee classes, we separated them into training set (542 tweets) and test set (53 tweets). We vectorized the tweets by both tf-idf and SVD (k =100) to use as our inputs. For classification, we used K nearest neighbor (KNN) algorithm on the vetorized tweets. We used k = 3 as our number of neighbors need for voting, brute force as the search algorithm and cosine similarity as the similarity metric. We got an overall accuracy of 86.79% from tf-idf vectors and 92.45% from SVD. Our classifier can detect “help/rescue” with 92.6%, “damage/aftermath” with 90% and “casualty” with 100% accuracy.