**Problem:**
With the advent of automated ticket purchasing came an introduction of a new market: the resale ticket market. Unlike standard venue ticket purchasing, ticket prices on resale sites, such as StubHub, are volatile and can be very expensive at certain times. This introduces the problem of deciding when to purchase concert resale tickets, since customers are not sure if the price will go up or down in the near future.

**Solution:**
Our project predicts whether the price of a concert ticket will go up or down over the next two days based on previous concerts' data, helping the buyer decide if they should buy the ticket now or wait and buy it later. We tracked a collection of general admission concerts in different cities, and recorded information about them, such as the ticket price, the city the concert is happening in, days till performance and twitter followers of the artist. We expected that the price of a concert ticket would fluctuate similarly for concerts with similar attributes, meaning that our data would naturally be clustered into concerts with the same classification label ('Buy' or 'Wait). So we therefore chose to implement Nearest Neighbor as our primary learning technique and we used 10 fold cross validation for our evaluation.

**Key Results:**
We performed 10 fold cross validation on our training set to calculate ZeroR, which we found to be 74.359%. Unfortunately, none of the learning techniques we tried could surpass ZeroR's accuracy. 10 Nearest Neighbor also achieved 74.359% accuracy. After calculating the linear regression correlation coefficient for each attribute, we found that none of the values were greater than 0.02, which explains why none of our learning techniques could surpass ZeroR's accuracy. Therefore, although it intuitively seems as though our attributes would affect the resale price of a concert ticket, our data proves to be irrelevant and our task too difficult.

***INSERT GRAPH & CAPTION HERE***

**<u>Detailed Project Report</u>**
Our data set consists of 9 total attributes:

- The current price of the concert ticket
- The retail price of the concert ticket
- The city the concert is in
- The number of days until the concert (from when the ticket price for this specific data point was checked)
- If the artist tweeted that day
- Whether the artist tweeted specifically about a concert that day
- The artist's twitter follower count (essentially measures his/her popularity)
- The concert venue's capacity
- The data points classification, which is binary—either to 'Buy' the ticket now, or 'Wait' and buy the ticket later

Each data point represents the price of a specific concert ticket on a specific date (so we have multiple data points for the same concert with each data point corresponding to a different current ticket price on a different date). We collected our data for the current ticket price from

songkick.com, a website that watches several other retail ticket websites (such as stubhub.com and ticketnetwork.com) and finds the current cheapest ticket price for a given concert. We normalized our artist twitter follower count and concert venue capacity attributes so that they wouldn't skew the distance function for Nearest Neighbor.

Our data set consists of a total of 117 examples. For evaluation, we performed 10 fold cross validation by breaking the 117 examples into 10 sets of about 11 examples. We then trained on 9 of these sets and tested on one (so one subset served as our validation set). We repeated this 10 times and then took the mean accuracy. As mentioned above, ZeroR achieved a 74.359% accuracy. We expected that the price of a concert ticket would fluctuate similarly for concerts with similar attributes, meaning that our data would naturally be clustered into concerts with the same classification label ('Buy' or 'Wait). So we therefore chose to implement Nearest Neighbor as our primary learning technique. We implemented Nearest Neighbor by coding the algorithm ourselves so that we could adjust the weight of each attribute as needed. One Nearest Neighbor achieved a 59.9915% accuracy, 3 Nearest Neighbor achieved a 64.9573% accuracy, 5 Nearest Neighbor achieved a 68.2308% accuracy and 10 Nearest Neighbor achieved a 74.359% accuracy. As you can see, Nearest Neighbor did not surpass ZeroR's accuracy. Since we had initially decided that implementing Nearest Neighbor made the most sense for our data set, we then proceeded to test a variety of other learning techniques we learned in class, using Weka, to try to figure out why Nearest Neighbor wasn't surpassing ZeroR's accuracy. Here are our accuracy results:
Bayes Net: 70.0855%
Naïve Bayes: 62.3932%
Logistic Regression: 62.3932%
Multilayer Perceptron: 54.7009%
Simple Logistic Regression: 62.3932%

None of the various learning techniques we implemented surpassed ZeroR's accuracy. We also tried implementing these learning techniques using a price ratio attribute (current price/retail price) in place of our current price and retail price attributes. But our accuracy results did not change by much:
ZeroR: 74.359%
One Nearest Neighbor: 53.8462%
3 Nearest Neighbor: 67.5214%
5 Nearest Neighbor: 65.812%
10 Nearest Neighbor: 74.359%
Bayes Net: 70.0855%
Naïve Bayes: 67.5214%
Logistic Regression: 62.3932%
Multilayer Perceptron 59.8291%
Simple Logistic Regression: 70.9402%

To better understand why none of the learning techniques we implemented were surpassing ZeroR's accuracy, we calculated the linear regression correlation coefficients for each attribute:
Current price: 0.0011
Retail price: 0.00074

Price ratio: 0.0153
Days till performance: 0.0028
City: 0.0018
Twitter followers: 0.00418
Venue Capacity: 0.0056

Since our attributes of if the artist tweeted that day and whether the artist tweeted specifically about a concert that day are both binary attributes, we used the Fisher Exact Test to calculate their correlation with our output. For if the artist tweeted that day, the Fisher Exact Test statistic value is 1. And for whether the artist tweeted specifically about a concert that day, the Fisher Exact Test statistic value is 0.668272. So neither of these values are small enough to conclude that these attributes and the output are not independent.

Based on our calculations that none of the correlation coefficients are greater than 0.02, we can conclude that even though it intuitively seems as though our attributes would affect the resale price of a concert ticket, there is no statistically significant relationship between any one attribute and our output. So it makes sense that none of the learning techniques we implemented surpassed ZeroR's accuracy. We also tried generating a decision tree from our data set, which we planned to use to help us determine the weights of each attribute in Nearest Neighbor's distance function. For each attribute, the earlier the Decision Tree splits on it, the higher that attribute's weight would be. However, Weka was unable to generate a decision tree of size greater than one since there is not a big enough correlation between our attributes and output.

Thus, we have concluded that the attributes we chose are not actually relevant and our learning task is too difficult. Since the price of a resale ticket is ultimately controlled by the individual person who posts that ticket on the resale website, there are too many reasons, many having to do with human nature, for us to account for, record and quantify in our data set. Also, it if often the case that when one person raises or lowers their resale ticket price, for whatever reason, other people tend to follow and do the same with their ticket prices. Since we did not have direct contact with the millions of individuals who are posting these tickets on resale ticket websites, it is hard to understand the exact motivation behind these trends. So moving forward, in order to continue with this project, one might do a more in depth exploration of these reasons and try to better understand when and why a resale ticket seller raises or lowers the price of his/her ticket on a resale ticket website.