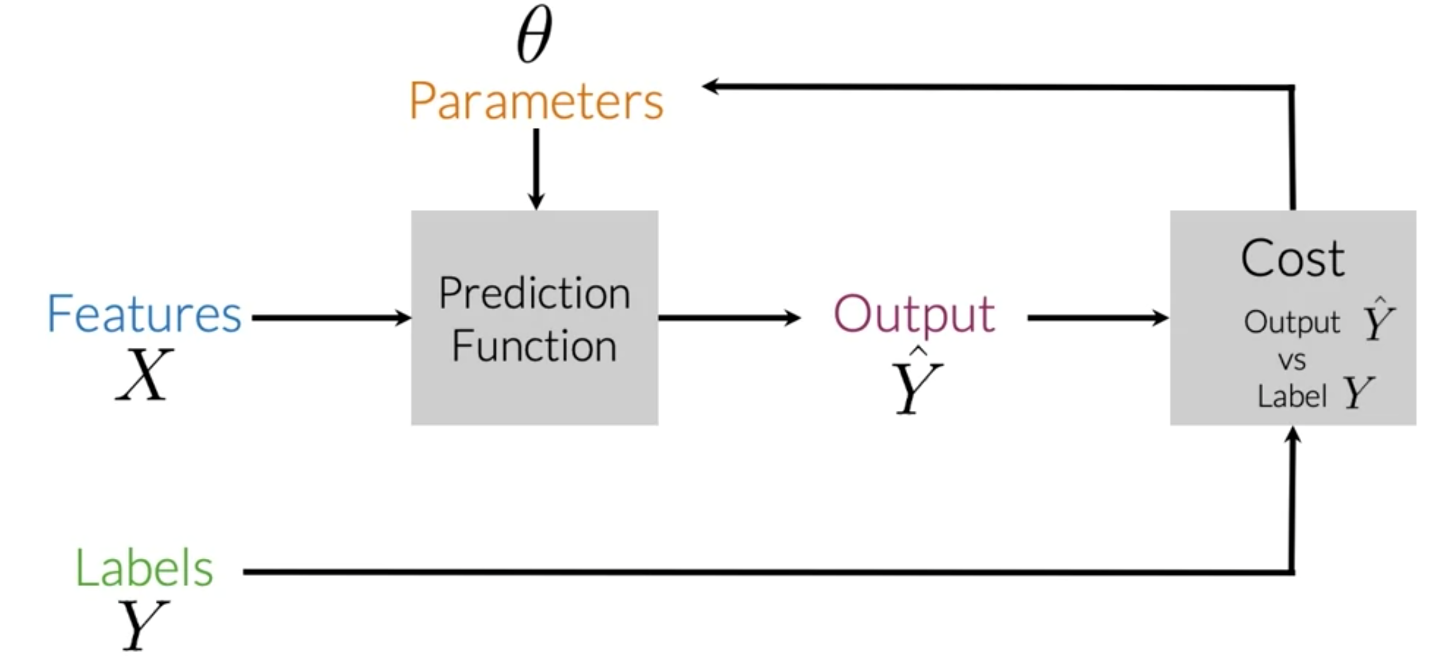
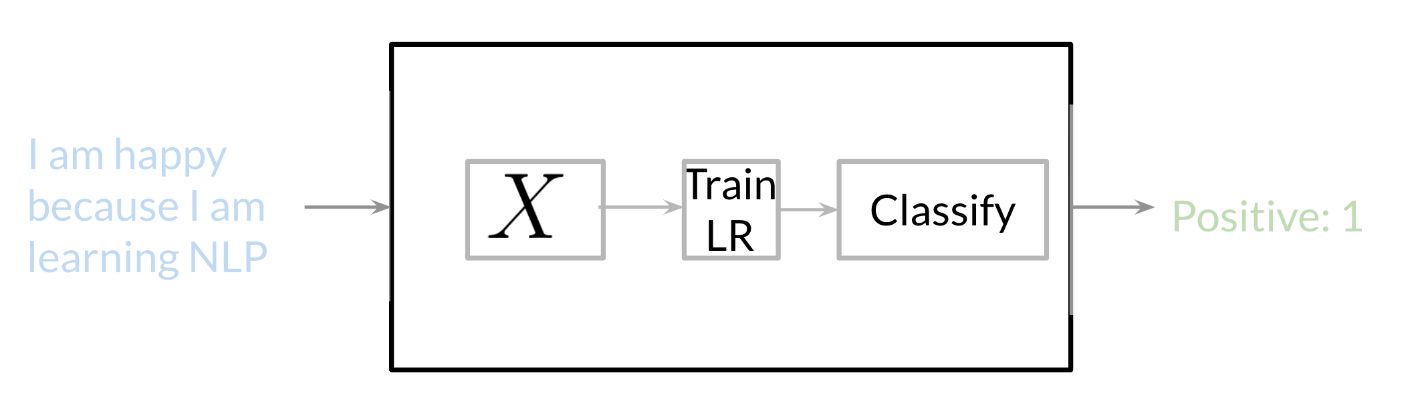
Supervised ML & Sentiment Analysis

In supervised machine learning, you usually have an input X*X*, which goes into your prediction function to get your \hat Y*Y*^. You can then compare your prediction with the true value Y*Y*. This gives you your cost which you use to update the parameters \theta*θ*. The following image, summarizes the process.



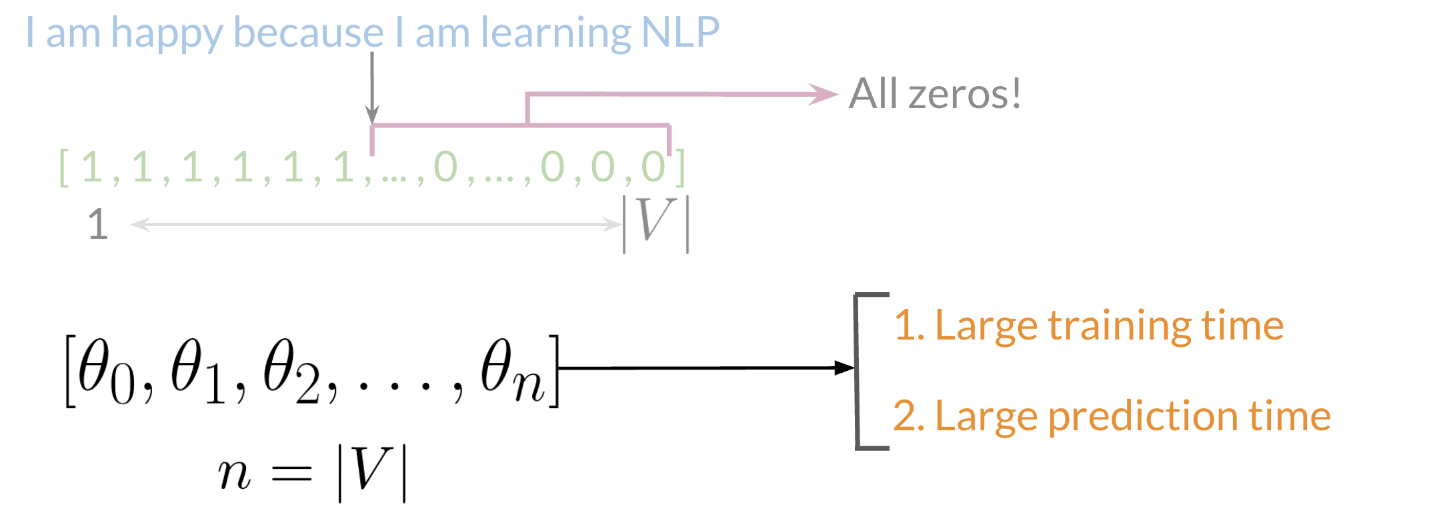
To perform sentiment analysis on a tweet, you first have to represent the text (i.e. "I am happy because I am learning NLP ") as features, you then train your logistic regression classifier, and then you can use it to classify the text.



Note that in this case, you either classify 1, for a positive sentiment, or 0, for a negative sentiment.

Vocabulary & Feature Extraction

Given a tweet, or some text, you can represent it as a vector of dimension V*V*, where V*V* corresponds to your vocabulary size. If you had the tweet "I am happy because I am learning NLP", then you would put a 1 in the corresponding index for any word in the tweet, and a 0 otherwise.



As you can see, as V*V* gets larger, the vector becomes more sparse. Furthermore, we end up having many more features and end up training \theta*θ* V*V* parameters. This could result in larger training time, and large prediction time.