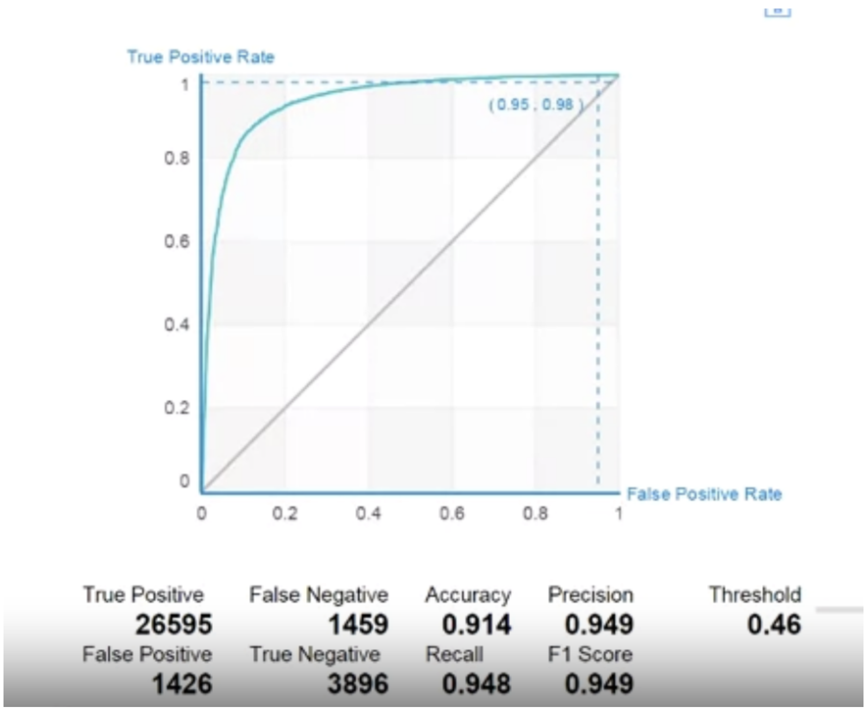
Data Exploration: Analyzing Product Sentiment	Read some product review data (product) - product.head()
Creating Word Count Vector	Add a new word_count column that counts the number of words in the review column - Output: count how many times each words appear in the review
Find Most Popular Product	1. Create a histogram via product.show() to see the most frequent item 2. Create a dataset to contain the reviews of only the most popular product 3. Order the created dataset in order of ratings and use .show() to see which one has the highest frequency 4. Can determine if this popular product has good or bad reviews
Sentiment classification (positive or negative)	1. Assign the ranks a. Unclear Ratings = 3 we ignore for now b. Positive Ratings = 4 or 5, value is 1 for true c. Negative Ratings= 2 or 1 value is 0 for false 2. Put the ratings of 1 or 0 in a new column called sentiment
Train the sentiment classifier	1. Assign train data and test data using *random_split* with 80% for training and 20% for testing 2. Use *logistic_classifier.create()*(*data _set, target, features, validation_set)* a. Data set = train data b. Target = name of column containing target variable (sentiment) c. Features = name of column containing features (word count) d. Validation_set = test_data 3. Output should show the number of iterations and the accuracy increasing with each iteration
Evaluate the Classifier with ROC curve.	1. ROC_Curve is a way to explore false positive/negatives 2. Use .evaluate(dataset, metric) a. Dataset = test_data b. metric=roc_curve 3. Use .show(view=’Evaluation”) to visualize the .evaluate outputs a. Output should show a curved line that evaluates false/true positives/negatives ROC Curves - Shows the probability of a true positive or false positive based on where the data hits on the curve.

	- Ex: value where FP=0.05 shows 0.2 chance of being a true positive  True Positive Rate 1 0.8 0.6 0.4 0.2 0 (0.95, 0.98) False Positive Rate 0 0.2 0.4 0.6 0.8 1 True Positive 26595 False Negative 1459 Accuracy 0.914 Precision 0.949 Threshold 0.46 False Positive 1426 True Negative 3896 Recall 0.948 F1 Score 0.949
Applying the Model	Applying the learned model to understand sentiment for the most popular product (Giraffe) 1. Extract the predicted sentiment for giraffe reviews from the sentiment_model into a new dataset called giraffe_reviews 2. Sort the reviews based on the predicted sentiment and explore using giraffe_reviews.sort('sorted item',ascending=false)
Exploring the most positive/negative aspect of a product	Dataset[0][column] => first time on the table Dataset[1]][column] => first time on the table . . . Dataset[-1]][column] => last item on the table Dataset[-2]][column] => second to last item on the table