Sentiment analysis report – Christopher Kong

1) The dataset used was the file Datafiniti_Amazon_Consumer_Reviews_of_Amazon_Products_May19.csv. It contains a lot of details and information of amazon product reviews, though we are interested only in the reviews text column. There are 28332 non-blank reviews.

2) We first use pandas module to read the csv file into a variable, then use the dropna function to remove any entries with blank review text. We then extract only the reviews.text column into a the main list we work with.

In the sentiment analysis function, we first use the en_core_web_sm model to analyze the review text, then use the sentiment function on it. The additional instructions of the assignment suggest doing text cleaning with functions like is_stop, lower and strip, but the examples provided in the spacy textblob website sentiment analysis section did not perform text cleaning, so to my understanding that is not necessary.

3) My program uses sentiment analysis on 10 random reviews every time it is run, and is successful at doing so. It first prints the basic information (row number and contents of the review), then the results of the sentiment function (polarity and subjectivity), then outputs whether the review is likely positive or negative, based on the polarity returned. From running the program a few times, the sentiment function seems quite accurate, and most of the reviews seem positive.

4) The program can quickly compile a lot of data and return the sentiment of each review, so it can quickly determine how well a product is received, or if people are likely to leave positive or negative reviews sitewide. However, the uses the simpler en_sm model, so its results may not be as accurate as the en_md model, though using the md model will take a longer time to compute.