3. For grouping similar literary items together based on subject topic we use Cosine Similarity metric. The reasons are given below:

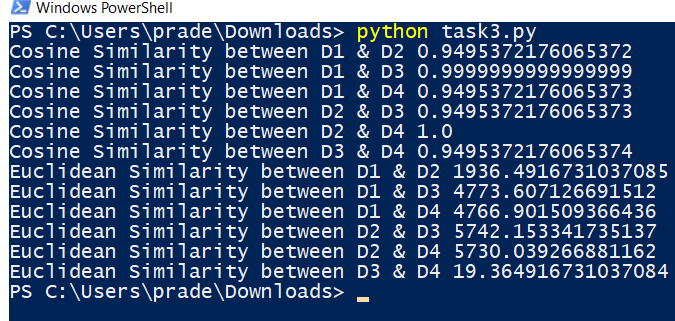
**Two** clusters are formed:

**Book on Data Science and Article on Data Science are similar. (1 and 3).**

**Book on Soccer and Article on Soccer are similar (2 and 4)**

They are evaluated and checked by both Cosine Similarity and the Euclidean Distance metrics.

* **Cosine Similarity metric:**
* Cosine Similarity is calculated using only the dot product and magnitude of each vector, and is therefore affected only by the terms the two vectors have in common,
* The cosine similarity is better at catching the semantic of each text, the direction the text points can be thought as its meaning, so texts with similar meanings will be similar.
* Cosine thus has some meaningful semantics for ranking similar documents, based on mutual term frequency.
* An important property of the cosine similarity is its independence of document length.. This happens for example when working with text data represented by word counts. We could assume that when a word occurs more frequent in document 1 than it does in document 2, that document 1 is more related to the topic. This happens when we are working with documents of uneven lengths.



* **Euclidean distance :**

Euclidean has a term for every dimension which is not zero in either vector. Another reason is that when modeling texts as vectors you will have more dimensions, the Euclidean distance metric is not good for high dimensional data.

Cosine similarity is generally used as a metric for measuring distance when the magnitude of the vectors does not matter. This happens for example when working with text data represented by word counts. We could assume that when a word (e.g. science) occurs more frequent in document 1 than it does in document 2, that document 1 is more related to the topic of science. However, it could also be the case that we are working with documents of uneven lengths. Then, science probably occurred more in document 1 just because it was way longer than document 2. Cosine similarity provides a correction for this.

You might also apply cosine similarity for other cases where some properties of the instances make so that the weights might be larger without meaning anything different.