COMP42315: Programming for Data Science Report

Anonymous Marking Code: Z0182576 Date: Februrary 2023

1 Question 1

The home page is loaded using the provided link and the "Publications" button is found by checking the contents of certain divs on the page. If the text is equivalent to "PUBLICATIONS", the relative link stored in the button is extracted and searched. The absolute URL for the Publications page is spliced together using a slice of the provided link and the relative link.

The Publications page is scraped for all the links to the topic pages in the topic navigation bar. I use a "flag" that updates its value when a message characteristic of the needed div is found, i.e. "Topic". Then, the link to each topic page is stored in a list.

The relative link to every publication page is scraped from each topic page, and the absolute URL to each publication page is made. Some publications are present on both pages so duplicate links are erased from the list using the `set()` function and then the result is converted back to a list called `pub_links`. Then, the topics belonging to each page are stored in a dictionary for later use.

Finally, `pub_links` is crawled and the citation number from each page is obtained using the `fetchCitationNumber` function and stored in a dictionary along with the title of the publication. The dictionary sorted into a list of tuples using the `sorted()` function which is then displayed.

Position	Title of publication	Number of Citations
1	A Quadruple Diffusion Convolutional Recurrent Network for Human Motion Prediction	228
2	Interaction Patches for Multi-Character Animation	201
3	Real-Time Posture Reconstruction for Microsoft Kinect	150
4	Simulating Competitive Interactions using Singly Captured Motions	112
5	Simulating Multiple Character Interactions with Collaborative and Adversarial Goals	112
6	Simulating Interactions of Avatars in High Dimensional State Space	102
7	Kinect Posture Reconstruction based on a Local Mixture of Gaussian Process Models	98
8	Interaction-based Human Activity Comparison	88
9	Environment-aware Real-Time Crowd Control	79
10	Real-time Physical Modelling of Character Movements with Microsoft Kinect	68
11	Posture Reconstruction Using Kinect with a Probabilistic Model	62
12	Interactive Formation Control in Complex Environments	61
13	A Two-Stream Recurrent Network for Skeleton-Based Human Interaction Recognition	60
14	Bi-projection based Foreground-aware Omnidirectional Depth Prediction	60
15	Filtered Pose Graph for Efficient Kinect Pose Reconstruction	55
16	Interpreting Deep Learning based Cerebral Palsy Prediction with Channel Attention	55
17	Emulating Human Perception of Motion Similarity	51
18	Differential Evolution Algorithm as a Tool for Optimal Feature Subset Selection in Motor Imagery EEG	50
19	Motion Adaptation for Humanoid Robots in Constrained Environments	50
20	Spatio-temporal Manifold Learning for Human Motions via Long-horizon Modeling	48
21	Arbitrary View Action Recognition via Transfer Dictionary Learning on Synthetic Training Data	45
22	Validation of an Ergonomic Assessment Method using Kinect Data in Real Workplace Conditions	45
23	Multi-layer Lattice Model for Real-Time Dynamic Character Deformation	44
24	DurLAR: A High-fidelity 128-channel LiDAR Dataset with Panoramic Ambientand Reflectivity Imagery for Multi-modal Autonomous Driving Applications	44
25	SkillVis: A Visualization Tool for Boxing Skill Assessment	44

Figure 1: Top 25 publications by citation number (in descending order)

2 Question 2

A class "Publication" is defined that will store important properties, such as title and year of publication, for ease and clarity. A list called `pub_page_objects` is made that will store these publication objects that are created when iterating through `pub_links`.

The `div` containing the LDOs is found by checking that the text inside starts with the string "Links".

Furthermore, the topics for each publication is taken from the dictionary defined in Q1 and their names are added to a list. Once all the publication objects have been created, I iterate through `pub_page_objects` to collate details on the year of publication, and the total number of LDOs and publications for that year. Using the number of LDOs and publications per year, the average number of LDOs per publication in each year is calculated and stored in a dictionary. The resulting dictionary is sorted by year in ascending order, and the results are graphed in the following scatter chart.

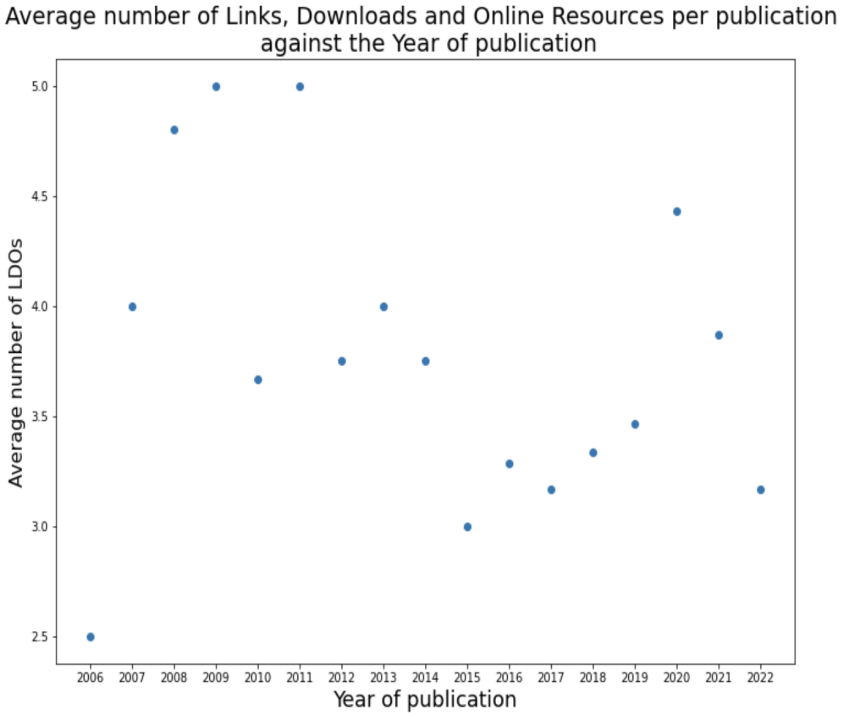


Figure 2: Average number of links, downloads and online resources per publication against the year of publication.

To obtain the top 25 publications by LDO count, the `sorted` function is used to generate a list of tuples by iterating through every publication object from `pub_page_objects` and sorting by its `number_LDOs` property returned using a lambda function. Then the resulting list is sliced to obtain only the first 25 results.

Position	Title of publication	Number of LDOs
1	Interaction Patches for Multi-Character Animation	8
2	Simulating Multiple Character Interactions with Collaborative and Adversarial Goals	6
3	Simulating Interactions of Avatars in High Dimensional State Space	6
4	Makeup Style Transfer on Low-quality Images with Weighted Multi-scale Attention	6
5	Foreground-aware Dense Depth Estimation for 360 Images	6
6	Resolving Hand-Object Occlusion for Mixed Reality with Joint Deep Learning and Model Optimization	6
7	DSPP: Deep Shape and Pose Priors of Humans	6
8	Physically-based Character Control in Low Dimensional Space	5
9	Interaction-based Human Activity Comparison	5
10	Fast Accelerometer-Based Motion Recognition with a Dual Buffer Framework	5
11	360 Depth Estimation in the Wild - The Depth360 Dataset and the SegFuse Network	5
12	Real-Time Posture Reconstruction for Microsoft Kinect	5
13	A Two-Stream Recurrent Network for Skeleton-Based Human Interaction Recognition	5
14	Interpreting Deep Learning based Cerebral Palsy Prediction with Channel Attention	5
15	Sparse Metric-based Mesh Saliency	5
16	Single Sketch Image based 3D Car Shape Reconstruction with Deep Learning and Lazy Learning	5
17	Angular Momentum Guided Motion Concatenation	5
18	Data-Driven Crowd Motion Control with Multi-touch Gestures	5
19	Semantics-STGCNN: A Semantics-guided Spatial-Temporal Graph Convolutional Network for Multi-class Trajectory Prediction	5
20	A Unified Deep Metric Representation for Mesh Saliency Detection and Non-rigid Shape Matching	5
21	Topology Aware Data-Driven Inverse Kinematics	5
22	Facial Reshaping Operator for Controllable Face Beautification	5
23	A Quadruple Diffusion Convolutional Recurrent Network for Human Motion Prediction	5
24	Formation Control for UAVs Using a Flux Guided Approach	5
25	Motion Adaptation for Humanoid Robots in Constrained Environments	4

Figure 3: Top 25 publications by LDO count (in descending order).

3 Question 3

3.1 Question 3a

The list `author_objects` is populated with `Author` class instances which contain information such as their name and the number of times they have been cited.

Position	Name	Times Cited
1	Hubert P. H. Shum	3266
2	Edmond S. L. Ho	1140
3	Taku Komura	1052
4	Howard Leung	655
5	Shuntaro Yamazaki	527
6	Qianhui Men	369
7	Shigeo Morishima	261
8	Nauman Aslam	260
9	Longzhi Yang	236
10	Shu Takagi	220

Figure 4: Top 10 authors by citation count (in descending order).

The number of articles per topic (from the top 10 cited authors) is provided in a bar chart as the topics are categorical and the number of publications is discrete.

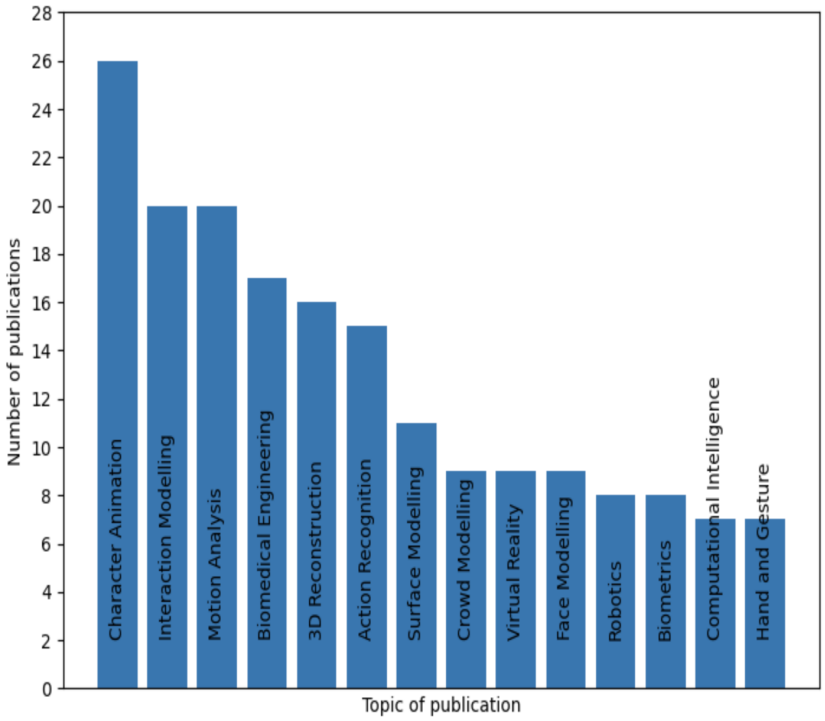


Figure 5: Number of publications per topic (of the publications by the top 10 most cited authors).

The number of times authors collaborate as coauthors in the publications based on topics is stored in the `topic_coauthors` dictionary, which prints for each top 10 author in a loop, but it is not shown here since it is very large in size.

To determine the most popular topics where authors collaborate, the average number of authors per publication for each topic was calculated.

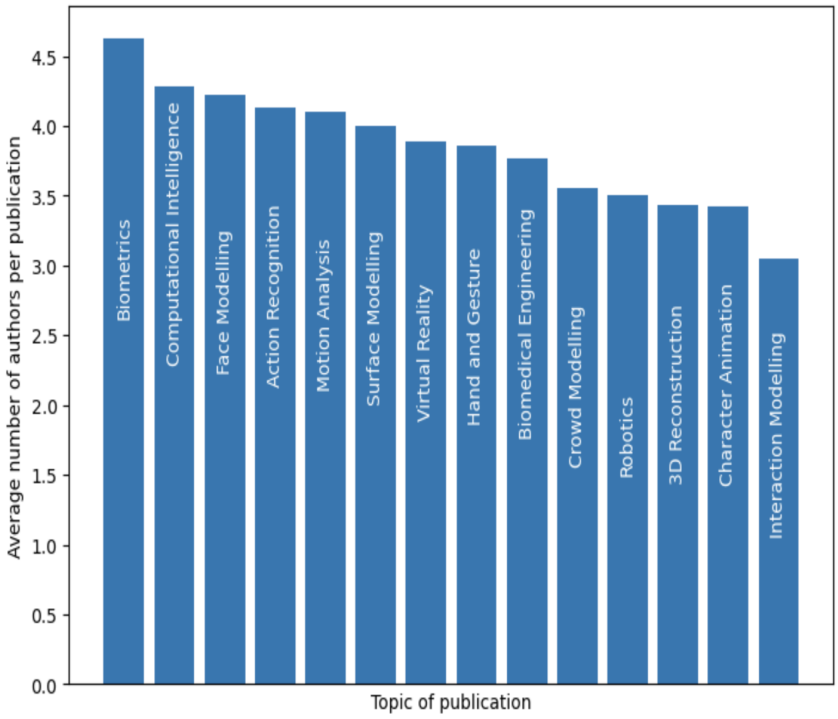
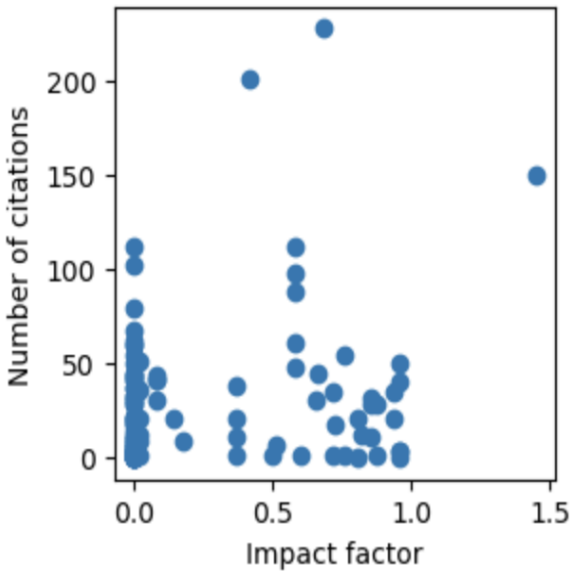


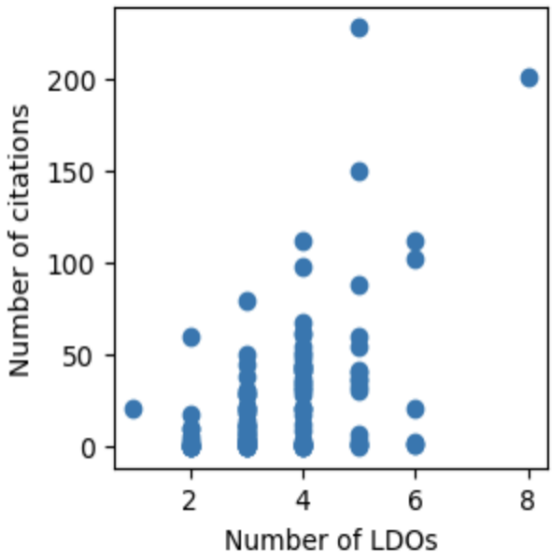
Figure 6: Average number of authors per publication per topic (of the publications by the top 10 most cited authors).

3.2 Question 3b

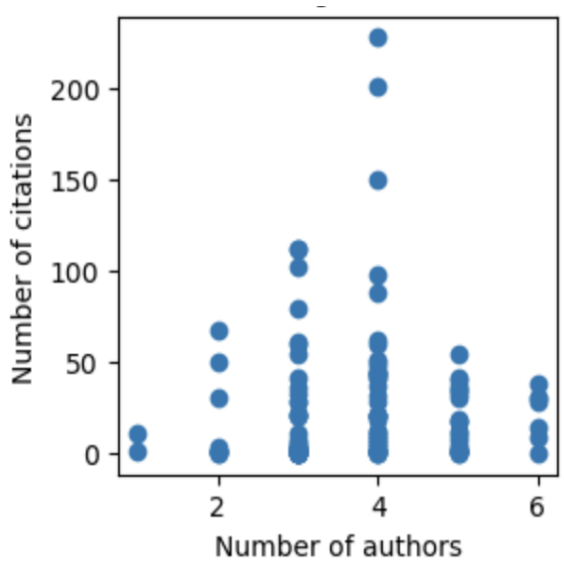
The publications of the top 10 authors are searched and their properties are compared with the citation count for that article, plotted on a scatter graph, and the product-moment correlation coefficient (PMCC) is calculated.



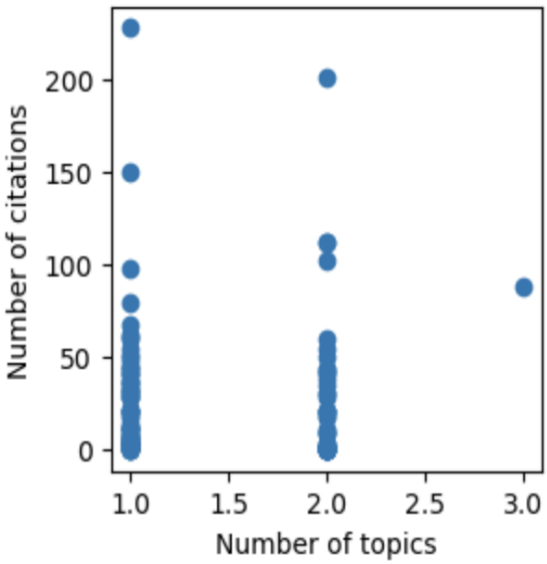
(a) PMCC = 0.26285531



(b) PMCC = 0.46685658



(c) PMCC = 0.03895068



(d) PMCC = 0.00132953

Figure 7: Plotted against number of citations of the publications by the top 10 most cited authors, clockwise from top left: impact factor, number of LDOs, number of topics, number of authors.

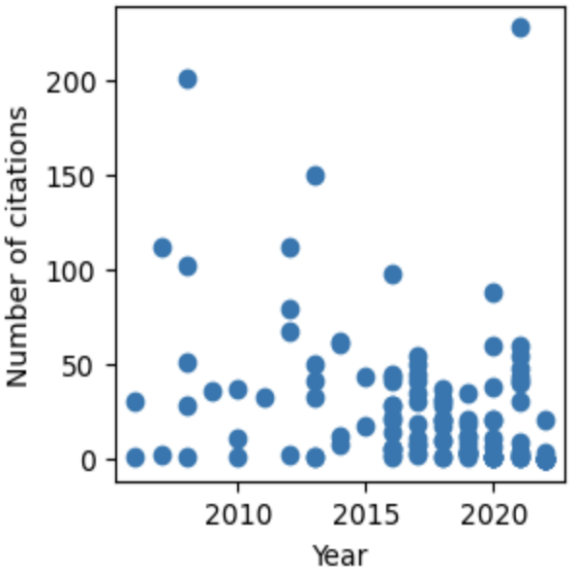


Figure 8: Number of citations in each publication each year (of the publications by the top 10 most cited authors). PMCC = -0.3260679

The number of LDO items is positively and more strongly correlated with citation count than the other metrics, whereas year is negatively correlated but has the second highest magnitude. For year, this is expected as newer articles will have had less time to be cited than older articles. Lastly, the topic is charted against the average number of citations in that topic.

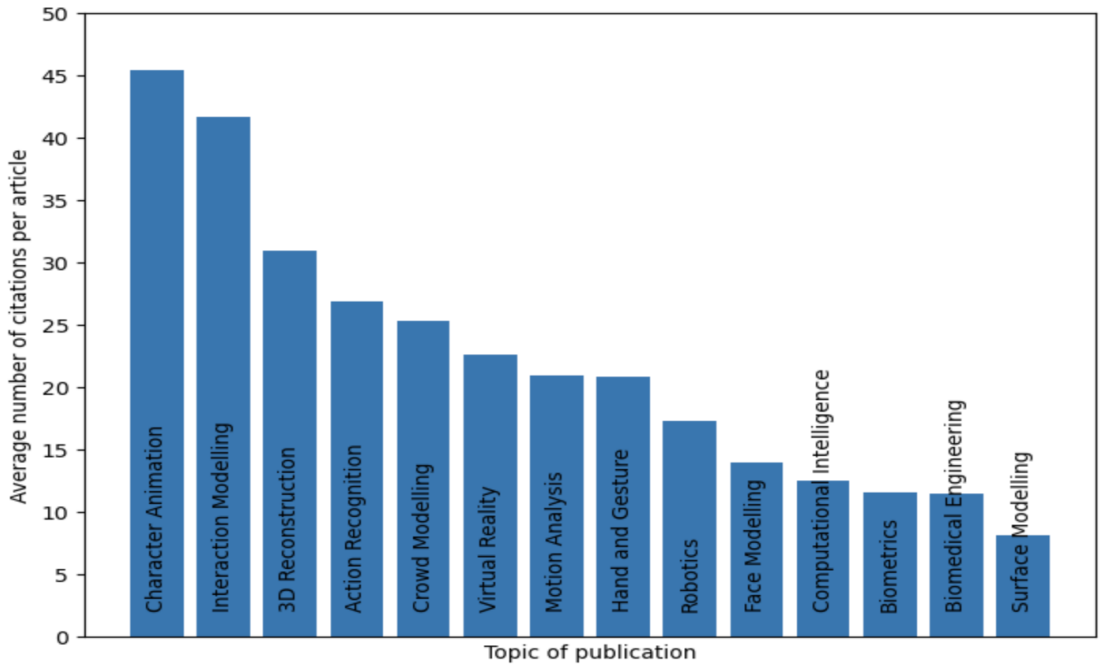


Figure 9: Average number of citations per publication in each topic (of the publications by the top 10 most cited authors).

Clearly, publications in topics such as Computer Animation and Interaction Modelling are more likely to be cited. It is fair to conclude that the number of LDO items, the year of publication, and the topic have the highest impact on citation count.

3.3 Question 3c

The type of venue is scraped from every publication in `pub_page_objects`. A dictionary stores the number of recorded impact factors with the type of venue. Only publications with venue type 'JOUR' (journal) have impact factors, while none of the others ('CONF', 'GENA', 'THES', 'Prep', 'CHAP') do. However, not every journal has an associated impact factor.

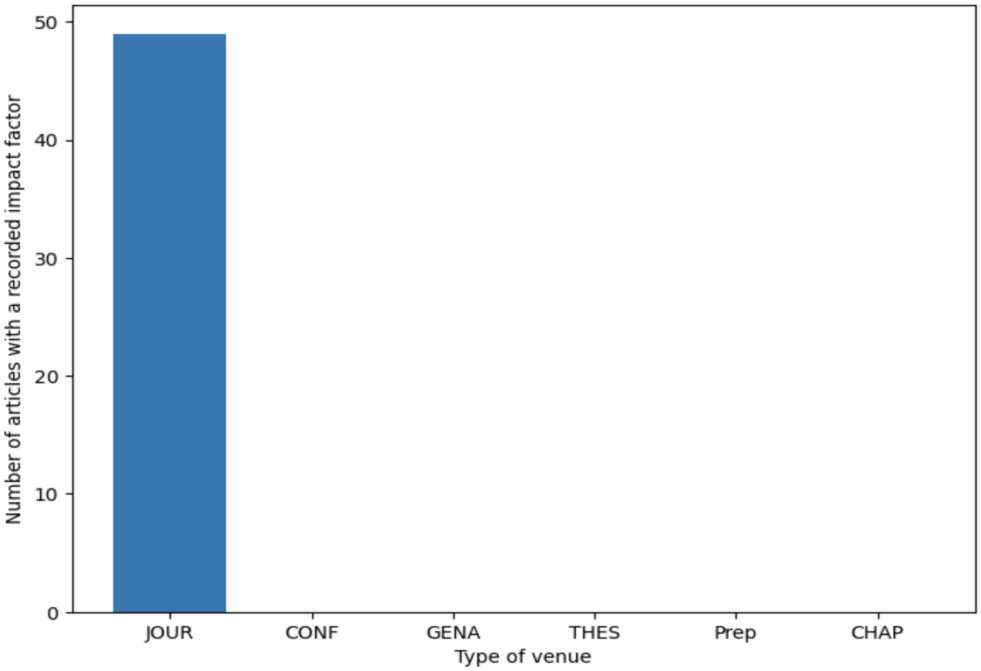
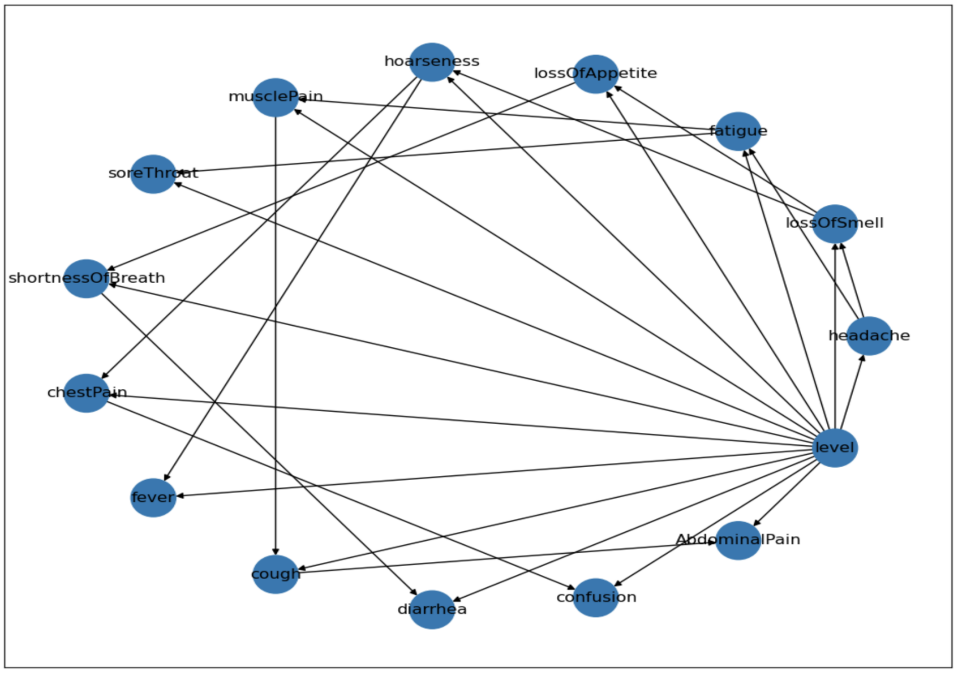


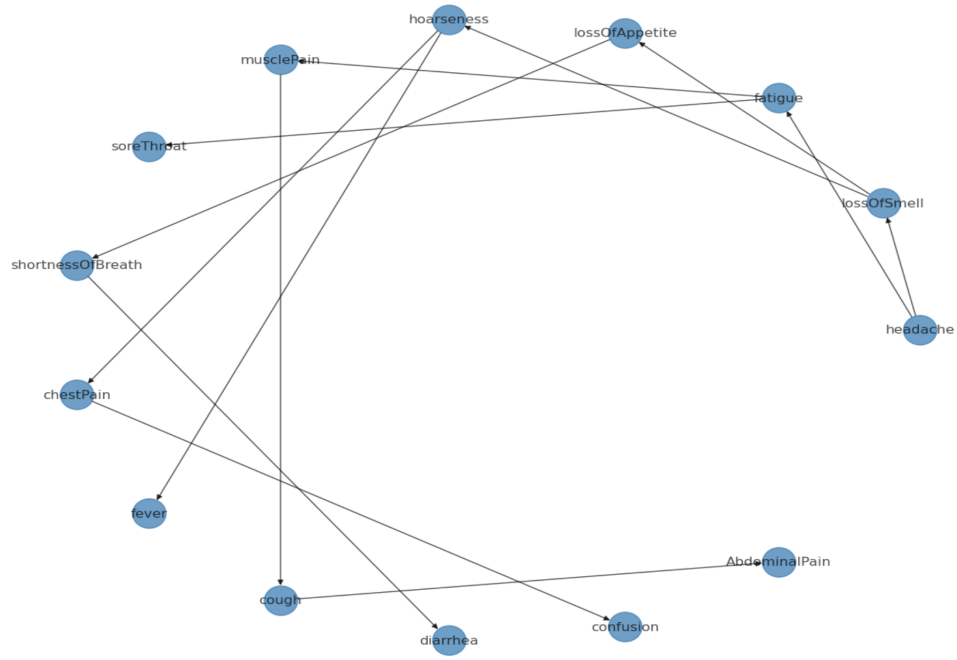
Figure 10: Number of articles with an associated impact factor for each type of venue.

4 Question 4

Some of the code used to answer this question is adapted from code provided at [AS]. First, the dataset was imported and the `id` column was removed. The data is checked for any missing entries (of which there are none), and the continuous variables are discretized using `discretize(df)`. To determine inter-dependencies between symptoms, two directed acyclic graphs (DAGs) was built using `TreeSearch` and the estimator type `tan`, since it allows for interactions between independent variables.



(a) With 'level' data



(b) Without 'level' data

Figure 11: DAGs showing the relationship between Covid-19 symptoms.

We see that the level of severity is directly related to the presence all symptoms. We also can see that fatigue is related to muscle pain, and that hoarseness is associated with chest pain. The tree graph was used to build a Bayesian model, which was tested to calculate the prediction accuracy for each symptom. The results, along with the average accuracy per symptom, were stored in a dictionary and are graphed below.

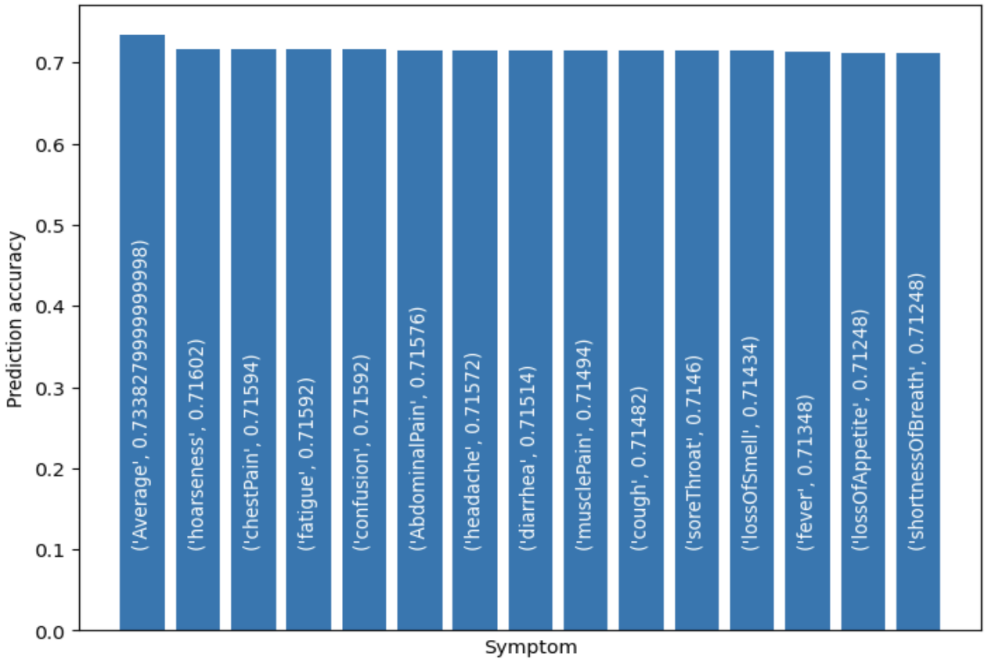


Figure 12: Prediction accuracy for each symptom

The conditional probability tables are too large to be presented in this report, but they indicate that those with a high severity of Covid-19 are probably experiencing most of the symptoms. Similarly, those suffering from a large number of Covid-19 symptoms are likely to be experiencing a harsher infection of Covid-19. Furthermore, looking at the conditional probability table for level of severity below, we can see that about 57% of sufferers have asymptomatic/very mild Covid, while the probability of suffering from any other level of severity (1-6) appears to be roughly equal - although this may be further dependent on other unrecorded features such as age of patient.

CPT of level:

level(1. 0)	57.1186
level(2. 1)	7.14957
level(3. 2)	7.09424
level(4. 3)	7.13424
level(5. 4)	7.1709
level(6. 5)	7.12224
level(7. 6)	7.21024

Figure 13: Level of severity and probability

References

[AS] Gagangeet Aujla and Pak Ho Shum. Complex data analysis and visualisation. Programming for Data Science, Workshop 4.