



Genetic Analysis of Prostate Cancer with Computer Science Methods

Yuxuan Li¹

Master of Engineering

UCL Computer Science

Supervised by Dr. Shi Zhou

Submission date: 1st May 2021

¹**Disclaimer:** This report is submitted as part requirement for the Computer Science Master's Degree at UCL. It is substantially the result of my own work except where explicitly indicated in the text. *Either:* The report may be freely copied and distributed provided the source is explicitly acknowledged

Or:

The report will be distributed to the internal and external examiners, but thereafter may not be copied or distributed except with permission from the author.

Abstract

Metastatic prostate cancer is one of the most common cancers in men. In the advanced stages of prostate cancer, tumours can metastasise to other tissues in the body, which is fatal. In this thesis, we performed a genetic analysis of prostate cancer tumours at different metastatic sites using data science, machine learning and topological network analysis methods. We presented a general procedure for pre-processing gene expression datasets and pre-filtering significant genes by analytical methods. We then used machine learning models for further key gene filtering and secondary site tumour classification. Finally, we performed gene co-expression network analysis and community detection on samples from different prostate cancer secondary site types. In this work, 13 of the 14,379 genes were selected as the most metastatic prostate cancer related genes, achieving approximately 92% accuracy under cross-validation. In addition, we provide preliminary insights into the co-expression patterns of genes in gene co-expression networks.

Contents

1	Introduction	2
1.1	Problem Outline	2
1.2	Contribution	3
1.3	Novel Methodology	4
2	Background	5
2.1	Prostate Cancer	5
2.2	Gene Expression	6
2.3	Machine learning	6
2.3.1	XGBoost	6
2.3.2	Model Evaluation Metrics	7
2.4	Network Science	8
2.4.1	Topological Complex Network	8
2.4.2	Community Detection	8
2.4.3	Gene Co-expression Network (GCN)	9
3	Project overview	10
3.1	Project plan	10
3.2	Dataset	10
3.3	Tools	11
4	Literature Review	12
4.1	Cancer Classification and Prediction	12
4.1.1	Analytical Methods	12
4.1.2	Machine Learning Methods	12
4.2	Gene Co-expression Network	13
5	Data Pre-process and Analysis	15
5.1	Dataset Cleansing	15
5.2	Dataset Analysis	16
5.3	Pearson Correlation Coefficient	17

5.4	Mask Selection Method	21
5.5	Normalisation	23
5.6	Gene Selection Strategies	26
5.7	Summary	28
6	Machine Learning Classification	30
6.1	XGBoost Model	30
6.2	Data Balancing	31
6.3	Key Genes Selection	32
6.4	Classification Result Evaluation	35
7	Gene Co-expression Network and Community Analysis	38
7.1	Gene Co-expression Network (GCN)	38
7.1.1	Intuition of Gene Co-expression Network	38
7.1.2	Dataset	39
7.1.3	Network Construction	39
7.2	Community Detection	41
7.2.1	Bone GCN	42
7.2.2	Liver GCN	43
7.2.3	LN GCN	43
7.2.4	Community Comparison Among Different GCNs	44
7.3	Evaluation	47
8	Discussion	49
8.1	Achievements	49
8.2	Methodological Contribution	49
8.2.1	Originality	49
8.2.2	Transferability	51
8.3	Limitation	51
9	Future Work	53
9.1	ML Models	53
9.2	Gene Co-expression Network	53
A	Model hyperparameters	59
B	Figures	60
B.1	Effective of mask selection method	60
B.2	Gene Co-expression Networks Visualisation	62
C	Code and Dataset	67

Chapter 1

Introduction

1.1 Problem Outline

Metastatic prostate cancer, or prostate cancer, is a type of cancer that arises from the prostate gland and is caused by uncontrollable growth of some prostate cells. Globally, it is the most commonly diagnosed cancer with the exception of breast cancer and is the 5th most deadly cancer among men [WHO14]. Prostate cancer is a malignant tumour in most cases and it can be difficult to control. Especially in the advanced stages of prostate cancer, the cancer cells can metastasise to other tissues in the body, such as the liver and lymph nodes, with the blood system and the lymphatic system. The cause of prostate cancer is largely unknown and there is no effective treatment to cure advanced prostate cancer. However, supportive treatment in the late stages can help patients control the progression of the disease and prolong their survival.

Prostate cancer typically develops very gently, and prostate cancer patients may be able to survive for years without symptoms or with mild symptoms. As the symptoms of prostate cancer are usually not obvious in the early stages, by the time a patient seeks treatment for severe symptoms, the patient is usually already at an advanced stage of prostate cancer. By this time the cancer has metastasised to other organs. For early stage prostate cancer, there is no effective way to know where the cancer will metastasise. If we can predict where the cancer will metastasise, we can use targeted treatments to prevent or mitigate cancer metastasis. This could significantly improve the survival rate of patients.

Recent medical research has shown that cancer is a genetic disease [Vog04] and many cancer research institutes around the world have collected a large amount of cancer-related data, such as gene expression data of cancer cells from different can-

cer patients. However, most of these data have not been well analysed and explored. Therefore, researchers can take advantage of data science and computer science to perform genetic analysis on cancers, which can potentially explore the relationship between genes and cancers.

In complex diseases such as prostate cancer, the search for key disease genes has become an important task in scientific research in recent years. There are 2 main methods for filtering, studying and analysing cancer-related key genes: machine learning methods and complex network analysis methods. In recent years, there have been many studies showing that some machine learning models based on decision trees, such as random forests, work very well in dealing with problems in the field of bioinformatics [Car06], and therefore such models have become popular in the field of bioinformatics. The use of support vector machine (SVM) [Cor95] models to predict cancer types from genetic alterations has also been found to be effective in much literature [Soh17] [Ram01]. Networked medical research using network science to abstract complex biological information into complex networks has also emerged as a typical disease research approach to understanding disease modules, identifying disease bio-markers and drug targets[Ire16].

1.2 Contribution

The main contributions of this project are threefold: firstly, we performed preliminary data cleaning and data analysis on a tumour cell gene expression dataset of prostate cancer patients. We proposed a general procedure for pre-processing gene expression datasets and pre-filtering of significant genes by purely analytical methods.

Secondly, we trained a high-performance decision tree-based machine learning model, XGBoost, to classify the site of metastasis in prostate cancer patients and achieved an accuracy of around 92% with cross validation. We also discovered two sets of key genes which are related to prostate cancer metastasis, one with 13 genes and the other with 34 genes.

Finally, we performed gene co-expression network analysis and community detection in prostate cancer patients. The results indicate that there is a strong gene community structure of prostate cancer-related genes, and these gene communities may play different roles, leading to cancer metastasis to different secondary sites.

1.3 Novel Methodology

Although much of the literature has been devoted to genetic analysis of cancer using pure analytical methods, machine learning methods and gene co-expression network analysis methods respectively, there has been very little work combining them all together. The most outstanding methodology in our work is to take advantage of the strengths of these three approaches and combine them together.

In this project, in order to reveal the mystery between prostate cancer metastasis and genetics, both the “answer” to the question and the “reason” behind the “answer” are important. We hope to be able to accurately predict the likely site of cancer metastasis based on the patient’s clinical data. But we also want to know which genes are factors in cancer and how they work together to influence the process of cancer metastasis. Because only by understanding the reasons can we have a better understanding in the life sciences.

The data analysis approach alone is not sufficient to achieve this goal, but it is useful for initially removing unrelated and unimportant information. The “black box” in machine learning models is suffering from interpretability issues [Jap98], and complex network models may not be as good at learning and prediction as machine learning models. In order to take advantage of each other’s strengths and overcome each other’s weaknesses, it is particularly important in this case that to ensemble machine learning and complex network models.

Chapter 2

Background

2.1 Prostate Cancer

Cancer is one of the most deadly types of disease that mankind has yet to fully conquer. Cancer forms when cells are genetically altered in such a way that they are able to proliferate rapidly and indefinitely. This uncontrolled proliferation causes the tumour to grow larger and larger, and at a later stage it can even invade a secondary site for tumour development through the blood and lymphatic systems. This process of tumour spread is called metastasis. After a tumour cell has metastasised, the cancer cells from the original site will attach to the new body site and continue to grow. So essentially, the type of cancer cells in the new site is still the same as the cancer cells in the original site. And the tumour at the new site is known as a secondary tumour. It implies a secondary tumour is composed of prostate tumours, rather than LN tumours, if the prostate cancer metastasises to the lymph nodes (LN). The tumour in the LN is then referred to as a metastatic prostate cancer in the LN, rather than an LN cancer. In the following part of this thesis, we refer to prostate cancer patients whose tumours have metastasised to the LN, bone and liver as “LN patients”, “Bone patients” and “Liver patients” respectively. And we also refer to prostate cancer secondary site tumour samples which at LN, bone and liver as “LN samples”, “Bone samples” and “Liver samples” respectively.

Prostate cancer is a type of cancer that arises from the prostate gland. It is the 5th most deadly cancer among men[WHO14].In 2018, 1.2 million people were diagnosed with prostate cancer and nearly 359 thousand died from it [Bra18] . It is a very common cancer in developed countries. Most prostate cancers grow slowly[NCI14], and may not cause any symptoms in the first few years. The greatest risk of prostate cancer is not the direct destruction of the prostate, but the possibility of the cancer rapidly metastasising to other body tissues and organs, particularly the bones

and lymph nodes[Rud15]. Metastatic prostate cancer can develop and grow rapidly, causing the organs in the secondary site to collapse quickly thus causing severe symptoms and hastening the patient’s death.

2.2 Gene Expression

The genes in DNA encode information about protein molecules that are critical to the body’s cells and perform functions essential to life. Different genes responsible for the production of different proteins, and different proteins perform different cellular functions in human body. In simple terms, to express a gene implies producing the corresponding protein for that gene. There are two main steps in the process of gene expression. First, the genetic messages in the DNA are transferred to the messenger RNA (mRNA) molecule by “transcription” [Liu16]. At this point, the resulting mRNA has a copy of the genetic information in the DNA. The cell then produces a protein molecule based on the genetic information on the mRNA by a process called “translation” [Bio14]. So, if want to know what kind of protein a cell will make and how much, it can be found out by measuring the type and number of genes covered in the mRNA. And this measurement process is known as “RNA sequencing” [RNA-Seq]. By sequencing, the kind of genes contained in the mRNA, and the “gene expression value” of each gene, can be obtained. Generally speaking, if a gene has a high gene expression value, then that gene is more expressed, which means that the protein corresponding to that gene will be produced in greater numbers.

Proteins expressed by a single gene usually do not have a specific function because of their simplicity. Often, multiple genes are involved in the synthesis of a functional and complex protein. The process of multiple genes working together to produce a complex protein is known as “gene co-expression”.

2.3 Machine learning

2.3.1 XGBoost

The decision tree based supervised machine learning algorithm XGBoost (Extreme Gradient Boosting) [XGB16] is a machine learning model that is often used in prediction tasks. This model has two outstanding features. One is that the model is

simple. The simplicity of the model makes it fast to train in the presence of large amounts of data. The second is that it is interpretable. The model can score the features of the data to identify the features that are important for prediction. This advantage is particularly important for this project because, in addition to wanting to know which cancer our model is predicting, we also want to know why this is the case and which genes are most important in determining the prediction. The Python “xgboost” package [XGBPy] implements the XGBoost algorithm for direct use and also provides a way to evaluate the importance of the features of the trained model.

2.3.2 Model Evaluation Metrics

(i) Confusion Matrix

In the task of evaluating the classification performance of a model, confusion matrix is a table which can show the detailed results of an algorithm’s predictions across multiple classes. In the confusion matrix, every single row of the confusion matrix stands for the amount of samples in the different classes classified by the model, and every single column stands for the amount of samples in the ground truth. For a perfect model that correctly predicts every instance, the numbers within the confusion matrix will be zero except for the diagonal.

(ii) Precision, Recall and F1 Score

Precision, recall and F1 score are the one of the most popular model evaluation metrics. Precision indicates to the percentage of true positive predictions in all number of positive predictions made by a predictive model. Precision metric measures how many the model’s predictions are relevant.

$$precision = \frac{\text{total number of true positive predictions}}{\text{total number of positive predictions}}$$

Recall indicates to the percentage of true positive predictions made by a predictive model in all number of ground truth positive instances. Precision metric refers how many relevant instances are correctly predicted.

$$\text{recall} = \frac{\text{total number of true positive predictions}}{\text{total number of actual positive instances}}$$

F1 is a measurement that takes into account both accuracy and recall. The F1 score ranges from 0 to 1 inclusive.

$$F1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

2.4 Network Science

2.4.1 Topological Complex Network

Network science uses topology, statistics and computer science to analyse how things relate to each other. In the field of network science, different instances are abstracted as nodes and connections between different instances are abstracted as edges (or links). It is particularly useful in the study of network representations of physical, biological and social phenomena. There are many useful and informative network features for network analysis. For example, the “degree distribution”, the probability distribution of node degrees across the network, which may reveal some of the network connectivity patterns behind the network. Another important network feature is the “node clustering coefficient”, which describes how a node’s neighbours are connected to each other.

2.4.2 Community Detection

In complex network analysis, community detection is a very useful and effective method for classifying nodes into multiple categories based on the relationships between them. In general, points within a community tend to closely connected to each other, and points between different communities tend to sparsely connected or

not connected. Communities can be either non-overlapping or overlapping. In non-overlapping communities, each node must and can only belong to one community, but in overlapping communities, each node belongs to at least one community and can belong to multiple communities.

2.4.3 Gene Co-expression Network (GCN)

Typically, functional proteins are very complex and their synthesis often involves the co-expression of multiple genes. The gene co-expression network (GCN) [Stu03] is a network used to represent gene co-expression information and relationships. In GCN, a point represents a gene, and if an obvious co-expression relationship exists between a pair of nodes, then a single undirected edge is used to connect them. The Pearson correlation coefficient can usually be used to measure the relationship between two gene co-expressions. Because co-expressed genes work together to synthesise a complex protein, the level of transcripts of co-expressed genes will increase and decrease together in different samples. GCNs can therefore be constructed from the correlation coefficients between genes. It is likely that there will be multiple obvious community structures in the GCN, where the genes in each community are likely to be co-expressed genes for certain proteins.

Chapter 3

Project overview

3.1 Project plan

The rest of this thesis is organised as follows. In Chapter 4, we did a series of literature reviews in the field of cancer gene analysis. The literature includes pure analytical methods, machine learning methods and complex network methods for cancer-related gene selection and cancer prediction. Chapters 5 to 7 are the main body of this project, explaining the methods and the analysis of the results. Chapter 5 is concerned with dataset cleaning and pre-processing. In this section, we performed a thorough analysis of the original dataset and then proposed a general approach to pre-process the gene expression dataset. Chapter 6 focuses on key gene filtering and an ML model classification. In this section, we trained and validated on a XGBoost machine learning classification model and filtered out 2 key gene sets that were highly correlated with prostate cancer metastasis. In Chapter 7, we constructed gene co-expression networks for LN, Bone and liver patients, respectively, and then performed network analysis and community detection on these networks. We also compared these networks with each other to obtain further conclusions. Chapter 8 is a discussion of the results achieved, novelty contributions and limitations, and Chapter 9 is a guide to future work.

3.2 Dataset

The dataset used in this project is Metastatic Prostate Adenocarcinoma (SU2C/PCF Dream Team, PNAS 2019), which can be downloaded at cBioPortal [cBioPortal], a public website for cancer genomics. The dataset consists of the expression values of 19,293 genes from 270 patients. All gene expression values were determined by RNA sequencing of FPKM [Soo11] standards.

Higher gene expression value of a gene means higher “gene expression” of this gene in a tumour sample. Gene expression value of “0” means the measured value is zero.

The 270 patients covered LN, Bone, Liver, Prostate, Lung, Adrenal and Other Soft tissues patients. In detail, There are 117 LN patients, 74 Bone patients, 40 Liver patients, 7 Prostate patients, 6 Lung patients, 1 Adrenal patient and 25 Other Soft tissues patients.

3.3 Tools

In this project, we use python 3.7, a very popular and powerful programming language, as the main programming language to implement our ideas. We use Jupyter notebook as the main development environment as it saves the results of our code runs in real time and is very convenient for viewing historical results. We use NetworkX, a Python package, in Chapter 7 to generate gene networks and then study the structure of these networks. We also use cdlib, another python packages which provides a wide range of complex network community detection algorithms, as a helper package especially for community detection tasks. Gephi, an open-source complex network analysis software is used in this project to visualize the networks and communities.

Chapter 4

Literature Review

4.1 Cancer Classification and Prediction

4.1.1 Analytical Methods

In 1999, Golub T et al. [Gol99] proposed an analytical method to differentiate acute myeloid leukaemia (AML) from acute lymphoblastic leukaemia (ALL) purely based on gene expression in bone marrow patient clinical samples. They analysed 6,817 human genes from bone marrow samples of 27 ALL and 11 AML patients and selected the top 50 most relevant genes based on the correlation between gene expression and disease type. Predictions were then made based on the differences in gene expression value between these genes in the two types of leukaemia. The results of this study proved in a mathematical way that the gene expression values of cancer-related genes differ between cancers and that it is feasible to classify cancers based on differences in the expression values of these genes. However, the study involved only 38 cases and their data covered just over 6,000 genes, which is limited compared to the total of over 20,000 genes in humans. They used only analytical methods, which may be effective when distinguishing only two cancers. But when it comes to multiple cancers classification, a purely mathematical analysis method alone may not be sufficient to solve the problem.

4.1.2 Machine Learning Methods

In 2001, Ramaswamy S et al. [Ram01] proposed a machine learning approach based on gene expression values of tumour samples to classify a wide range of cancer cells. This study involved 218 tumour samples and 90 samples of normal human cells, covering up to 14 different cancers, and 16,063 genes. They used a support vector

machine (SVM) algorithm to classify the samples and achieved an accuracy of 78%. This study further demonstrates that it is possible to differentiate and diagnose cancers by differences in gene expression values of relevant genes between different cancers. They also found that the genes associated with different cancers tend to vary and are not fully captured by a few dozens of genes. Even though a few genes can be used to predict most cases, prediction can be more difficult for two cancers with similar tumour cells.

This study was further improved by Soh et al. [Soh17] In this improved project, a larger dataset was used, which included 6640 samples involving 28 types of cancer. They used a support vector machine with recursive feature selection and eventually filtered out only 50 key genes and obtained an overall prediction accuracy of about 77.7%.

4.2 Gene Co-expression Network

As mentioned in the background section, information on gene co-expression is embedded in GCN, so a complex network analysis approach is also a way to explore cancer-related genes. In the work of Zhang, Bin and Horvath, Steve [Zha05], a general methodology for GCN analysis is proposed. This is a very important work as their proposed GCN has strong biological implication, which makes the results of the network analysis on this network valuable. This work focuses on the construction and feature analysis of GCNs. Details including the use of Pearson correlation as a co-expression measure for linkage weights, and then normalise edge weight values by applying different adjacency functions, including Hard Thresholding, sigmoid and power adjacency functions. They have also done preliminary work on gene co-expression community detection.

A recent work conducted by Savary, C et al. [Sav20] in 2020 used the construction of GCNs and community detection method to identify cancer-related genes and to investigate which group of genes correspond to which type of cancer. They explored cancer-associated genes with the dataset which contains gene expression values from 820 paediatric cancer samples across six cancer group types. They constructed a weighted GCN where the linkage weight between each two genes was the absolute value of the Pearson correlation coefficient between these two genes in the samples. They analysed the community structure of this network and then to find the communities associated with each cancer by matching the correlation of the communities to the tumour type. These gene communities were then explored in depth to further

filter out key genes associated with cancer.

Chapter 5

Data Pre-process and Analysis

5.1 Dataset Cleansing

The dataset used in this project consists of the expression values of 19,293 genes from 270 patients. The 270 patients covered cases of metastatic prostate cancer to the LN, bone, liver, prostate, lung, adrenal and other soft tissues. However, for prostate, lung and adrenal patients, there are only a few cases. In order to avoid the specificity of a small number of data cases, this work focuses only on studying the categories with the majority of cases, i.e. LN, Bone and Liver patients. Therefore, the data under study includes 231 tumour samples from 117 LN patients, 74 bone patients and 40 liver patients, and each row is the gene expression value of one gene in all 231 tumour samples, and each column is the expression value of 19,293 genes in one tumour sample from one patient.

Then, we did follow steps to cleansing the dataset:

- a. Remove any genes with gene expression values equals to 0 for all patients.
- b. Remove any duplicate genes.
- c. Reorder columns: Reorder the patients samples so that columns 1-117 are LN patients' sample, 118-191 are Bone patients' sample, and 192-231 are Liver patients' sample.
- d. All values are cut short to 3 digits after the decimal point.

After the primary data cleaning process, 14,379 different genes remained, and this dataset was the main dataset for this project study. It is also referred to as the “original dataset” in the rest of this thesis.

5.2 Dataset Analysis

To better understand the characteristics of the gene expression values for all genes, we need to perform a thorough data analysis of the original dataset. The analysis consisted of examining the gene expression values for each gene, and the distribution characteristics of the maximum, minimum, mean and median values across the 231 patients.

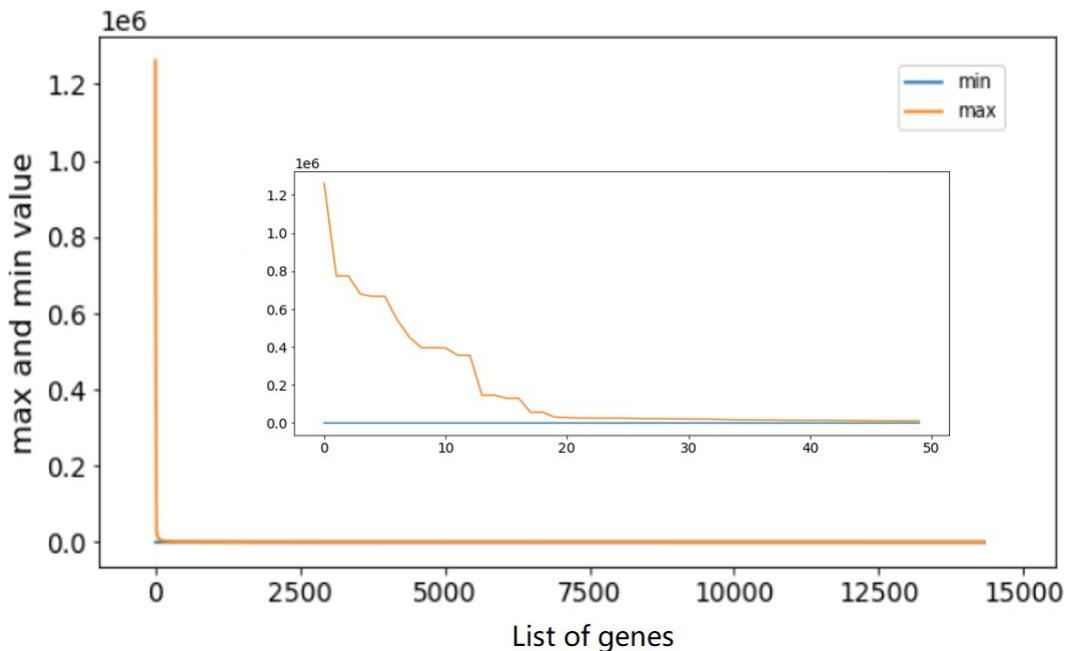


Figure 5.1: *The max and min gene expression values of the 231 patients for each of the 14,379 genes, where the genes are ordered in descending order of their Max value. The inner plot shows the top-50 ranked genes.*

Figure 5.1 shows the maximum and minimum expression values for each of the 14,379 genes in 231 patients. We can see that only about 20 genes had a very large difference between the maximum and minimum values. The range of gene expression values for most genes is relatively small. The difference between the maximum and minimum values of gene expression values is an informative indicator as it shows the sensitivity of a gene in all patients. In general, the higher the sensitivity of a gene, the easier it is for us to distinguish between patients with different cancer types. We then define the difference between the maximum and minimum gene expression values for a given gene as the “gene sensitivity” of that gene. As observed in Figure 5.2, most of the genes have gene sensitivity values in the range of 10 to 100. However, there are still some genes with extremely large or small gene sensitivities.

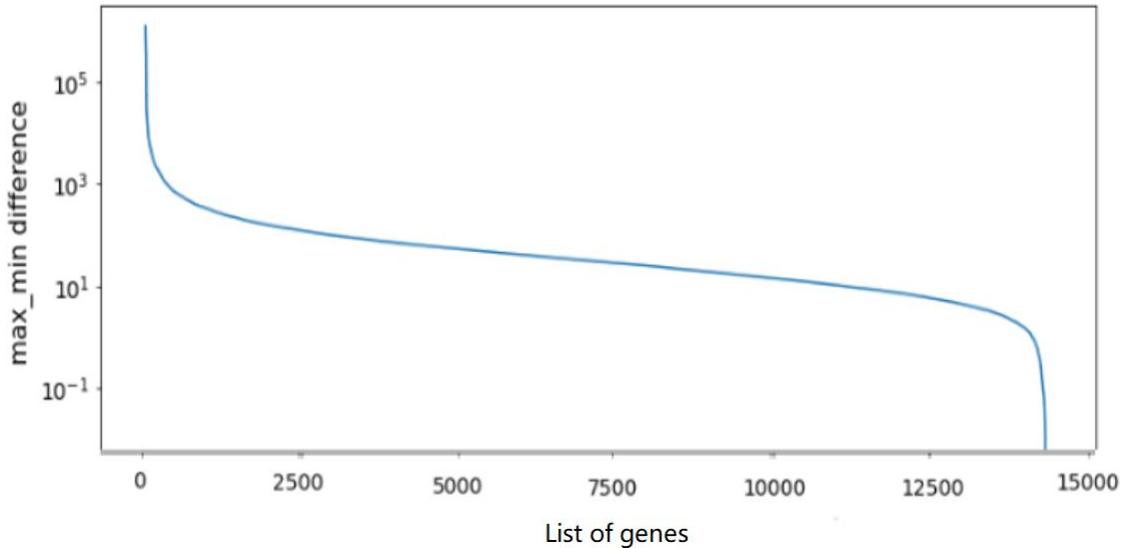


Figure 5.2: *The gene sensitivity of each gene, where genes are ordered in decreasing order of sensitivity value and sensitivity is in log scale.*

Other than gene sensitivity, we are also investigated in the gene expression value distribution among the 14,379 genes.

In Figure 5.3, we note that the mean values for almost all genes are greater than the median, indicating a very uneven distribution of gene expression values, with most patients having very small gene expression values. For example, for one gene, a few patients had very large gene expression values, which drove up the mean value, but many patients had very small gene expression values so that the median gave a smaller value. To validate this thought, we added Figure 5.4, a 3D plot of the distribution of gene values for all genes. Clearly, this graph verifies that for most genes, many patients have very small gene expression values and only a few patients have very large gene expression values.

At this point, we have a general idea of the data set. For most genes, gene expression sensitivity varies widely, and gene expression values are very unevenly distributed.

5.3 Pearson Correlation Coefficient

One of the goals of this project is to identify cancer-related genes. Biological knowledge suggests that co-expressed genes work together to synthesise a complex protein and that the transcript levels of two co-expressed genes rise and fall together in different samples. Therefore, a direct study of the values of gene expression values is not very helpful for this work. In combination with biological knowledge, we believe

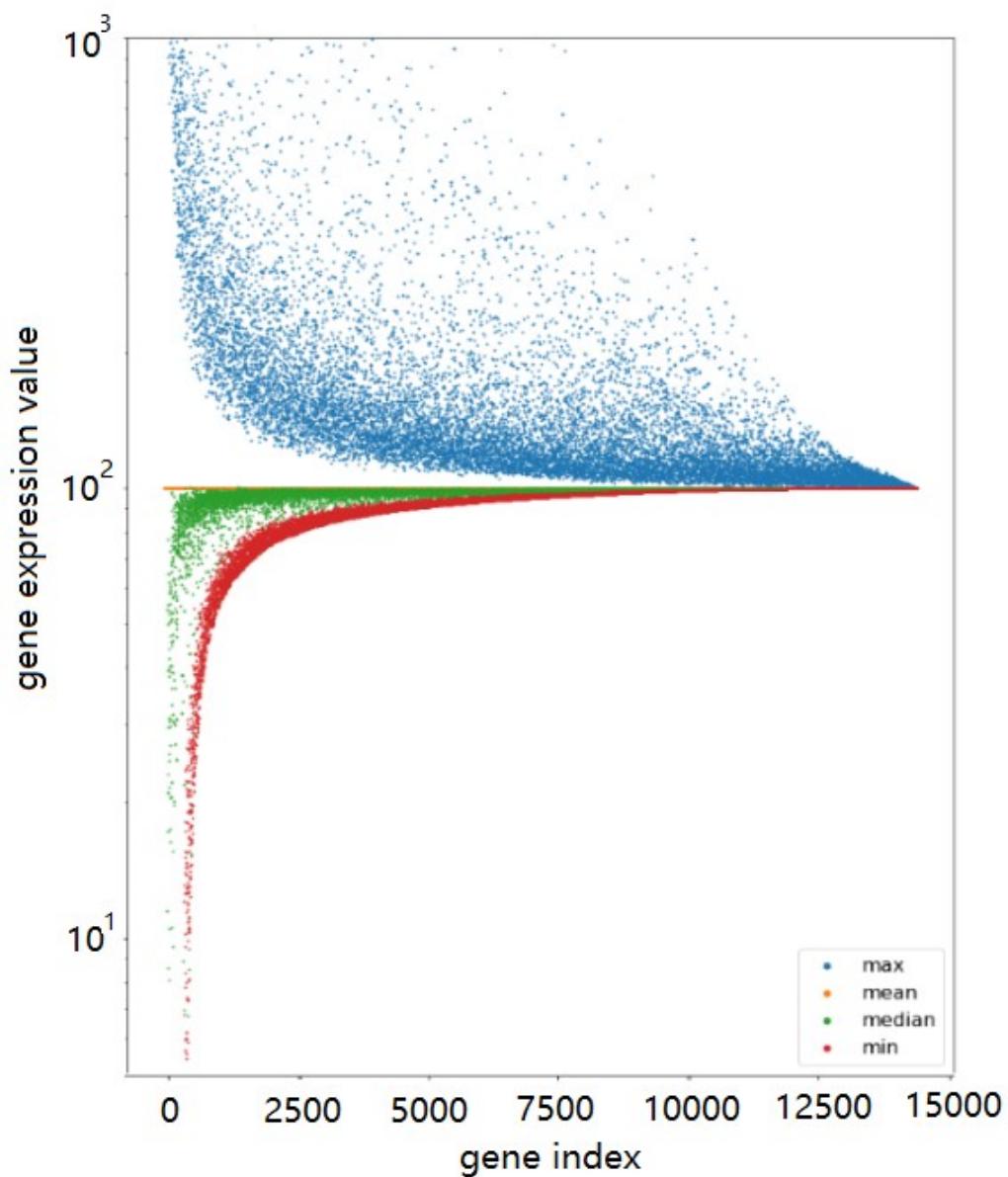


Figure 5.3: Maximum, mean, median and minimum gene expression values for 14379 genes in 231 patients. The genes are ordered in descending order of the mean value, Y-axis is in log scale.

that it would be more biologically meaningful to examine the trends and patterns in the rise and fall of gene expression values across samples. The Pearson correlation coefficient is an algorithm that describes the correlation between the fluctuating trends of two data. We therefore expected that the Pearson correlation coefficient for cancer-related gene expression values should be significantly higher for patients with the same type of prostate cancer secondary site than for patients with different types of prostate cancer secondary site.

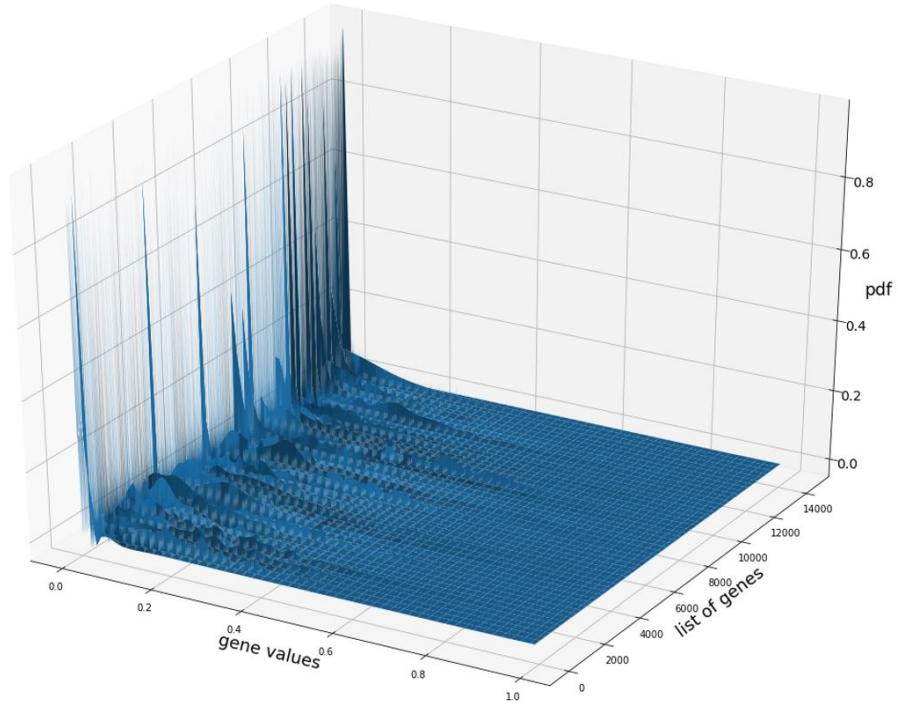


Figure 5.4: *Gene value probability distribution 3D plot: X: gene values linear mapping to range from 0 to 1 inclusive, and evenly divided into 100 bins; Y: all 14,379 genes; Z: the probability value of the gene Y. Genes are sorted in descending order of their sensitivity.*

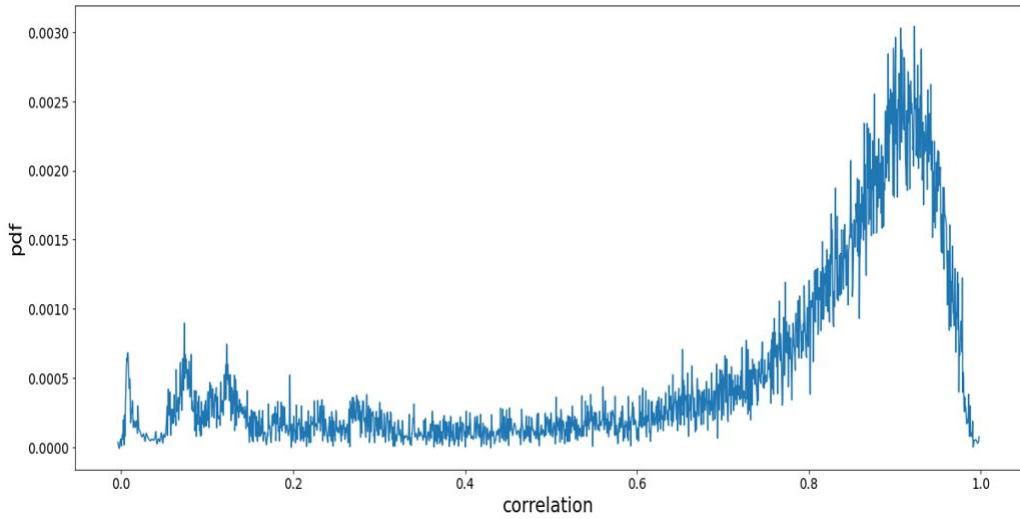


Figure 5.5: *Probability distribution function of Pearson correlation between all pairs of patients.*

Figure 5.5 and 5.6 show the probability distribution function and cumulative distribution function of the Pearson correlation between all patient pairs. We found that most of the patients were highly correlated with each other, and the results in Table 5.1 show that patients with the same type of cancer secondary site were

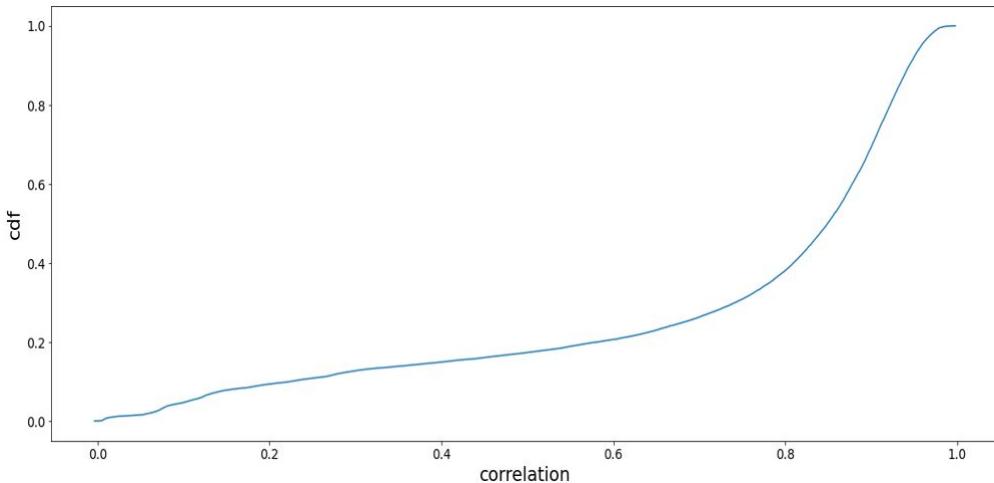


Figure 5.6: *Cumulative distribution function of Pearson correlation between all pairs of patients.*

Mean of Pearson Correlation	LN	Bone	Liver
LN	0.818	0.68	0.802
Bone		0.647	0.672
Liver			0.825

Table 5.1: *Mean value of Pearson Correlation between patients of the same type of cancer (diagonal) and between patients of different types of cancer (off-diagonal).*

highly correlated with each other, but patients with different types of cancer secondary site, such as LN and liver, were also highly correlated. This is also reflected in Figure 5.7, where red indicates a high positive correlation, blue indicates a high negative correlation and white indicates no significant correlation. The square areas on the diagonal lines are correlations between patients with the same type of cancer, and the rectangles on the non-diagonal lines are correlations between patients with different types of cancer. We see that all the areas are in red, which indicates that most patients are highly correlated with each other, regardless of whether they have the same or different types of cancer secondary site patients. So it is not likely to effectively distinguish between different patients just based on all these 14,379 genes. This is probably because most genes that are not associated with cancer metastasis, which play an important role in the calculation of correlation values, are taken into account. In addition, since all 231 patient tumour cell samples were essentially prostate cancer cells with similar gene expression values, even though the cancer metastases related genes were expressed at different values in patients with different cancer metastasis types, they were so few in number compared to other common genes that the overall correlation would ignore these differences.

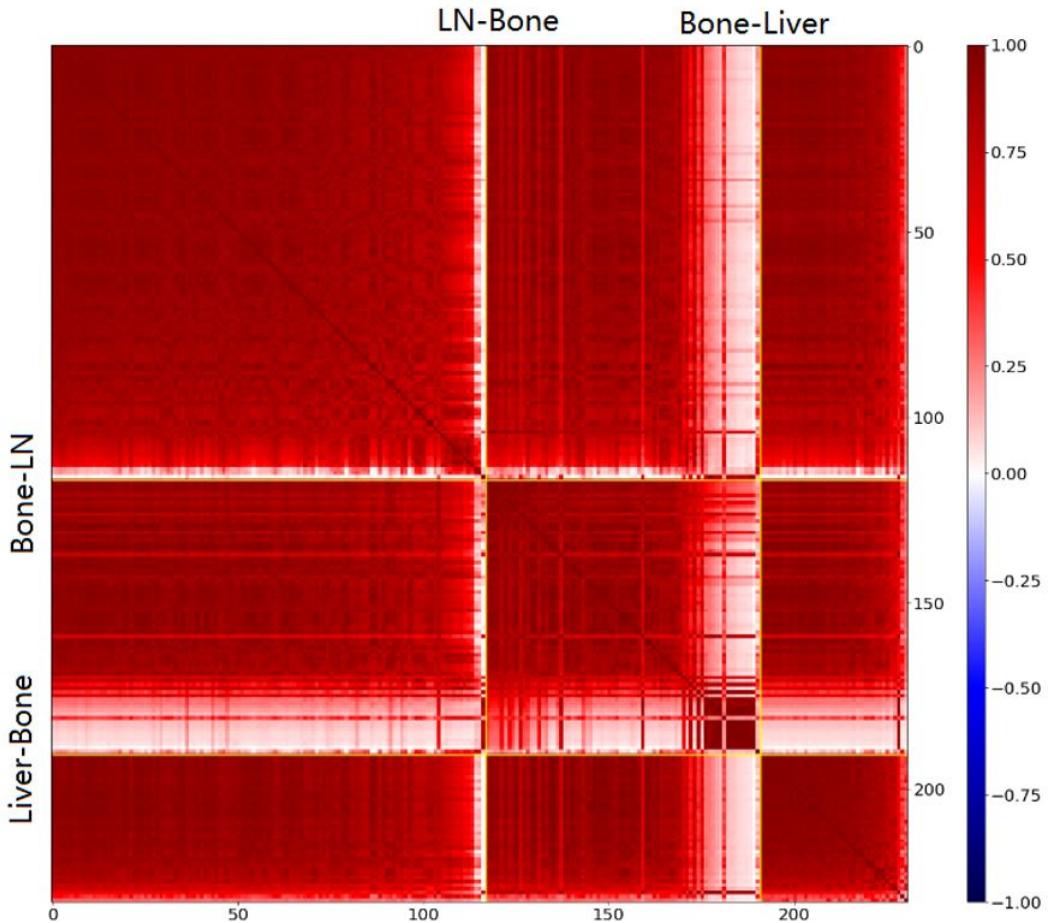


Figure 5.7: *Heatmap of Pearson coefficient of the 14,379 genes between the 231 patients. Patients of the same type of cancer are grouped together and sorted in descending order of their mean coefficient value within the group.*

Therefore, the next step should be to filter out the irrelevant genes and keep only the genes associated with these three types of cancer metastases. With only genes associated with cancer metastasis, we expect only the diagonal squares in the heat map to be red and the other regions to be blue or white.

5.4 Mask Selection Method

To find genes that may be associated with cancer metastasis, we propose the mask selection method. In our understanding of cancer genetics, cancer metastasis related genes will have higher (cancer-causing genes) or lower (cancer-suppressing genes) gene expression values in patients with that type of cancer. Based on this idea, to filter out LN metastases related genes, we created an “LN metastases mask” for this dataset: a list of 231 elements, with the first 117 elements being 1 and the others being 0 (remembering that, in the dataset, the first 117 columns are gene expression

values for LN patients). So, if a gene has a high positive correlation with the “LN metastases mask”, it is likely to be an LN site cancer-causing gene. And if a gene has a high negative correlation with the LN cancer mask, it is likely to be an LN cancer-suppressing gene. Similarly, we also created “Bone metastases mask” and “Liver metastases mask”.

LN metastases mask representation:

$$LN \text{ metastases mask} = [1, \dots, 1, \underbrace{0, \dots, 0}_{114 \text{ 0s}}]$$

Bone metastases mask representation:

$$Bone \text{ metastases mask} = [0, \dots, 0, \underbrace{1, \dots, 1}_{74 \text{ 1s}}, \underbrace{0, \dots, 0}_{40 \text{ 0s}}]$$

Liver metastases mask representation:

$$Liver \text{ metastases mask} = [0, \dots, 0, \underbrace{1, \dots, 1}_{40 \text{ 1s}}]$$

To demonstrate the effectiveness of the mask selection method, we plotted Figure 5.8. The results in Figure 5.8 show that the mask selection method is very effective, as we can easily select bone metastasis related genes based on this graph. We can clearly see that for genes with a high positive correlation with the bone metastasis mask, they have a low correlation or high negative correlation with the other two masks. And for genes with a high negative correlation with the bone metastasis mask, they have a low correlation or high positive correlation with the other two masks. These obvious gaps make them easy to distinguish.

The bone site cancer-causing genes are selected as follows: set a threshold value, e.g. 0.15, then any gene with a bone metastases mask correlation higher than 0.15 will be selected. Similarly, a threshold of -0.15 be set to select bone site cancer-suppressing genes in such a way that genes with the mask correlations below -0.15 would be selected. The same gene selection steps apply to LN and liver as well.

We have also sorted the above graph by descending order of correlation values with the LN and liver metastases masks respectively, which are put under Appendix section. All of these three plots give similar patterns. This suggests that we can use this masked selection method to pre-select genes that may be associated with prostate

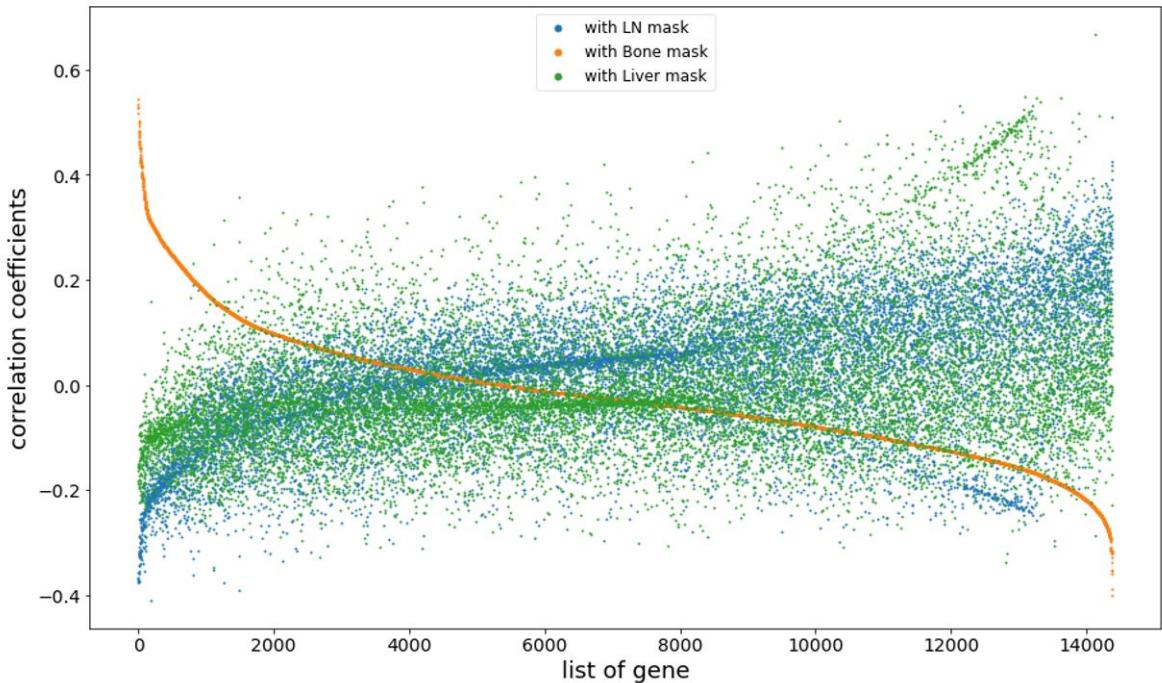


Figure 5.8: X are 14,379 genes and Y are 3 correlation values with each of the LN, Bone and Liver metastases masks in different colours. Sort the genes in descending order according to the correlation value with the Bone metastases mask.

cancer metastasis.

5.5 Normalisation

As concluded in the ‘Dataset Analysis’ section, large variation in gene expression sensitivity and the uneven distribution of gene expression in the original dataset may introduce extra difficulties for data analysis and potentially affect the effectiveness of the mask selection method. Therefore, additional data processing of the dataset is necessary. In this section, we have attempted several normalisation schemes for gene expression values. The normalisation schemes we tried, along with an explanation of each scheme, are listed in Table 5.2.

To verify and evaluate the performance of each normalization scheme, for gene values under each of the normalization schemes, do the following steps:

- a. For each normalised gene (each gene has 231 data, one for each patient sample), calculate the correlation with each of the “LN metastases mask”,

Normalisation scheme	Interpretation
Origin	Use the original gene expression values
Range	For each gene, linear scale the original gene expression values to [0,1].
Log	For each gene, take log ₁₀ value on all the 231 gene expression values, and then scale to range [0,1].
Rank	Calculate the gene sensitivity values across the 231 patients. Then replace values with ranks (between 1 and 231), same values have the same rank. Lastly scale the ranks into range [0,1].
Logit	Take logit value, i.e. $y = -\log(1/x - 1)$ on the range-normalised values
Logit-log	Take logit value, i.e. $y = -\log(1/x - 1)$ on the log-normalised values.

Table 5.2: *Normalisation scheme interpretation.*

“Bone metastases mask” and “Liver metastases mask”.

- b. Set a “threshold”, and if the absolute value of the correlation between a gene and any of the three masks is greater than or equal to the “threshold”, then the gene is retained. To find an optimal threshold, try thresholds in the range of 0.05 to 0.6 with a step size of 0.05.
- c. Fill a table to record the number of genes selected by the three metastases masks with different thresholds. Then plot a graph to reflect this data.

Based on the results, we found that 0.15 and 0.2 were ideal thresholds because for each normalization scheme, these two thresholds selected a sufficient number of genes, and the positive and negative thresholds selected a relatively balanced number of genes. If a loose threshold like 0.1 is used, the number of genes considered will be too large, making this gene selection less effective. However, if a higher threshold is used, for example a threshold of 0.3, the number of genes selected will drop sharply to about only 1000. This may exclude some potentially informative gene candidates.

In the early stage, we want to take more genes into consideration, so we firstly use threshold of 0.15 instead of 0.20. To find out which normalisation scheme was most effective, we plotted a series of patient correlation heatmaps in which we set the threshold to 0.15 and calculated the correlation between any two patients for each normalisation scheme using only the union of the genes selected by LN, Bone and Liver cancer mask.

Theoretically, if only cancer metastasis-related genes are involved, the ideal heat map would have only the diagonal squares in red and the other regions in blue or white. The results of the above experiments are added in Table 5.3. We can clearly

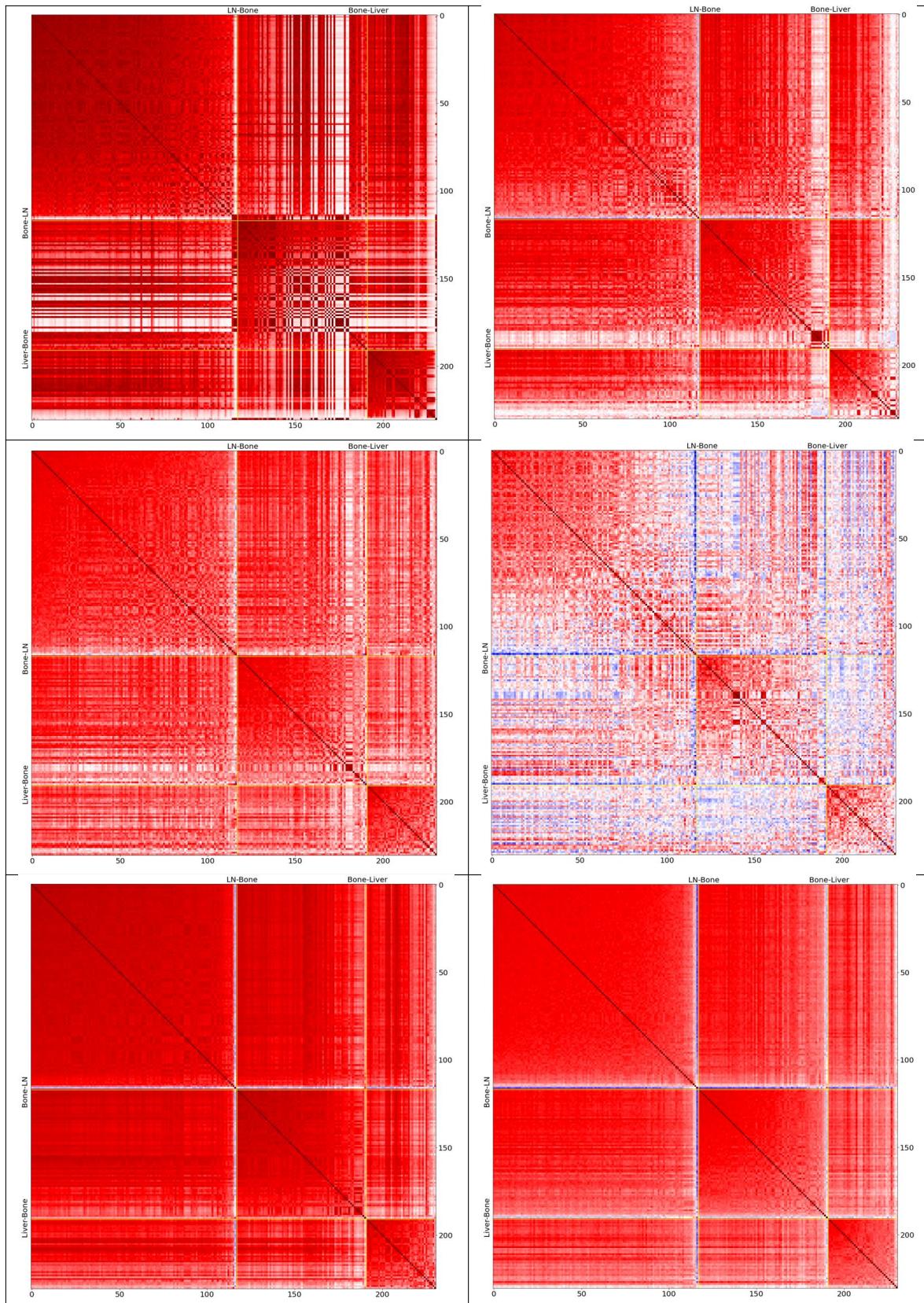


Table 5.3: Heatmaps of correlation between any of 231 patients' selected genes where correlation is calculated using different normalised values: None, Range, log, rank, logit and log-log normalisations, with threshold set to 0.15. Patients with the same kind of cancer are put in the same area and sorted in descending order of their mean coefficient value within the group.

see that in Table 5.3, the rank normalisation is the best normalisation scheme among the six schemes. Therefore, we will focus only on the rank normalisation of gene expression values.

5.6 Gene Selection Strategies

However, even with rank normalization and a threshold of 0.15, there is still too much red in the cross-section of the heatmap. This is probably because the threshold we used was too low, allowing many unrelated genes to be taken into account. Therefore, we should only consider genes that overlap because, as shown in Figure 5.8, if a gene is selected by a positive threshold of one cancer mask, it is likely to be selected by a negative threshold of another cancer mask as well. So the intersecting genes selected by the three masks are the ones that can be used to distinguish between different cancers. Based on this thought, we tried to use only the intersecting genes of the three mask selected, and refer these genes as “three masks intersect genes”.

The intersecting genes of the three masks selections with threshold of 0.15 gives 559 genes. These 559 genes are very informative and important genes, so we refer to these genes as SET_559. This gene set will be used later in the chapter “Gene co-expression Network and Community Detection”.

We plotted the correlation heatmap of patients with the SET_559 in Figure 5.9, we note that liver metastases are easily detected from other site metastases, but the correlation between LN and bone patients remains high, making it difficult to distinguish between LN and bone patients based on the 559 genes currently selected.

Therefore, we conducted the following experiments to further search for a set of genes that could be used to differentiate between LN and bone patients. First, we ignored all liver patients and focused only on distinguishing LN and Bone patients. This time, instead of setting the threshold to 0.15, we chose a higher threshold of 0.2 to increase the differentiation of genes. We then selected genes with high positive correlation with LN mask and high negative correlation with Bone mask, as well as genes with high negative correlation with LN mask and high positive correlation with Bone mask. In other words, we want to keep genes that are selected by both masks but give opposite correlation signs to both masks. The idea behind these experiments is that if a gene is highly positively (or negatively) correlated with both the LN metastases mask and the Bone metastases mask, then it is likely to be a

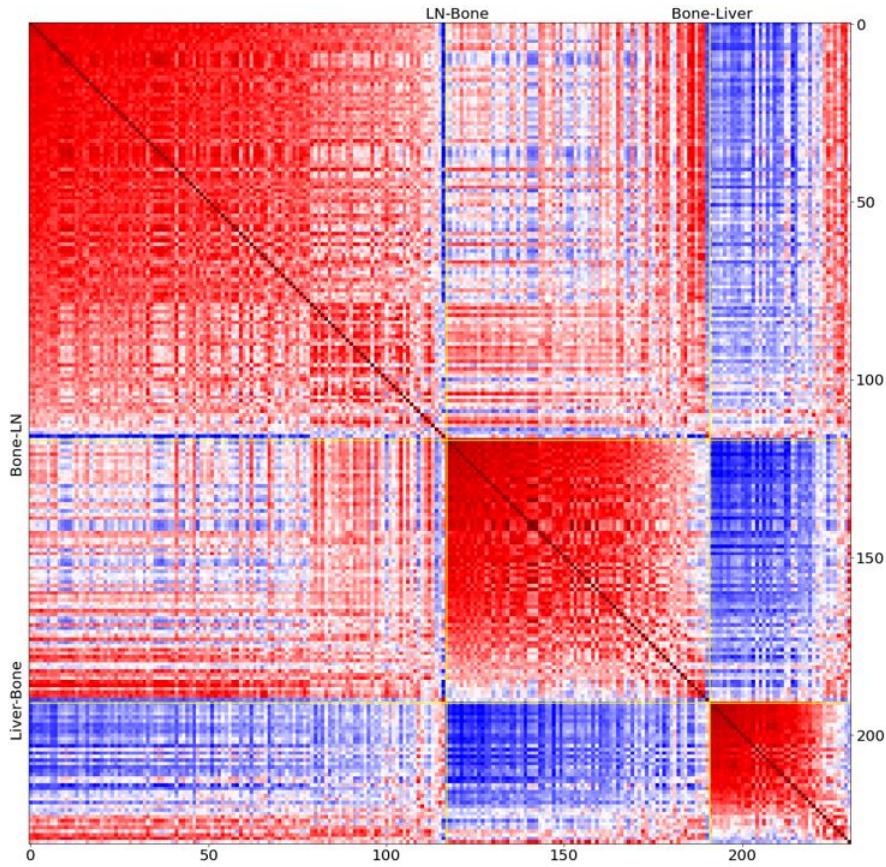


Figure 5.9: *Heatmaps of correlation between any of 231 patients' rank normalisations genes selected by 3 all of metastases masks with threshold set to 0.15. There are 559 genes are taken into account.*

common gene in individuals and not associated with cancer. So this additional gene selection processes will attempt to filter out these genes.

The results show that genes that are selected by both the LN and Bone masks but give opposite correlation signs (which we named “LN-Bone strong genes”) produce good results. Next, we consider only the intersecting genes of the “three masks intersect genes” and the “LN-bone strong genes” with mask selection threshold of 0.2.

Based on results in Fig 5.10, it is obvious intersecting genes of the “three masks intersect genes” and the “LN-bone strong genes” with mask selection threshold of 0.2 gives a more discriminating heatmap, which is what we would expect to see from the selected genes. This 133 genes set is also very informative and important gene set, so we refer to these genes as SET_133. As SET_133 is obtained from SET_559 with stricter constraints, SET_133 is a subset of SET_559. This gene set will be used later

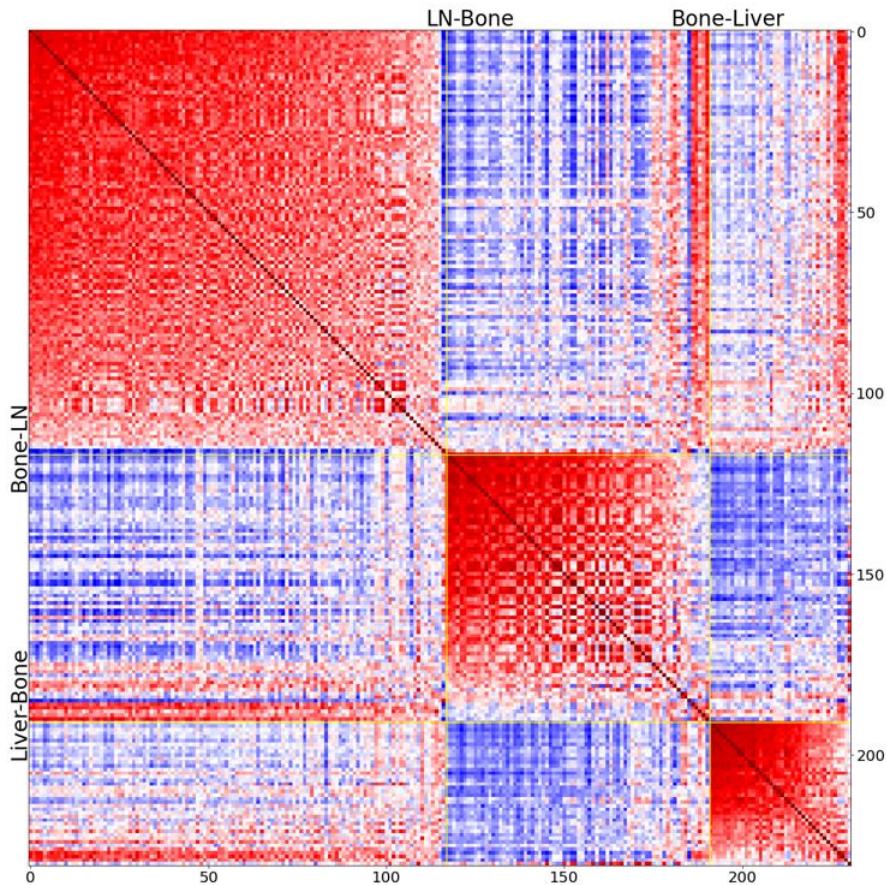


Figure 5.10: *Heatmaps of correlation between any of 231 patients' selected genes correlation is calculated using rank normalisation, with threshold set to 0.2. There are 133 genes considered which are intersect genes of "three masks intersect genes" and "LN-Bone strong genes"*

in the chapter "Machine Learning classification" and "Gene Co-expression Network and Community Analysis".

5.7 Summary

In this section, we did a though data analysis on the prostate cancer gene expression value dataset, and by a large amount of experiments, we find a effective method to pre-process and pre-select cancer related genes. In summary, the logic of gene selection process can be expressed in the following way:

- a. Do a data cleansing to remove obviously meaningless genes.
- b. Apply rank normalisation on gene expression dataset

- c. Select genes with mask selection method.

In this particular work, the mask selection method can be represented by:

$$\text{Selected gene set} = (|C_{Liver}| \geq t) \wedge (|C_{LN}| \geq t) \wedge (|C_{Bone}| \geq t) \wedge (C_{LN} * C_{Bone} < 0)$$

where t is the selection threshold, $t=0.2$, C_{LN} is the correlation of a gene with the LN metastases mask, C_{Bone} is the correlation of a gene with the Bone metastases mask, C_{Liver} is the correlation of a gene with the Liver metastases mask. After this process, only 133 out of 14,379 genes are selected.

Chapter 6

Machine Learning Classification

One of the aims of this project is to filter out genes related to prostate cancer metastasis and on this basis to classify prostate cancer metastasis secondary sites tumours for diagnostic purposes. Many literature has discussed the use of machine learning methods to classify and predict different cancer types based on differences in gene expression values. However, this project is challenging because essentially the dataset for this work is all prostate cancer cell samples, meaning that they are all cell samples from the same type of cancer. The only difference is that the samples are from different sites of secondary metastasis. This fact suggests that only certain few genes are associated with prostate cancer cell metastasis in these samples, while most genes are not associated with cancer metastasis at all. This requires us to build a sharp and sensitive model that can identify a small number of genes from thousands of genes. This is a challenging task for a machine learning model because the number of genes(features) is much larger than the number of tumour samples(datapoints). When the number of features is greater than the number of datapoints, machine learning models often suffer from the curse-of-dimensionality problem [Bel57] and the over-fitting problem [Tet95]. So to make the machine learning models avoid these problems and have a better performance, we did a gene pre-filter in the previous section. In this section, we use SET_133 as a start. That is, we only consider 133 genes from 231 samples in this section.

6.1 XGBoost Model

There are also many machine learning models available for classification and prediction tasks. However, we decided to use the supervised machine learning algorithm XGBoost (Extreme Gradient Boosting), which is based on decision trees. Although a complex model like a neural network model is powerful and can discover the under-

lying patterns behind the data, it suffers from an interpretability problem. It is like a “black box” that produces a single result when we throw data to it, but we have no way of knowing how it makes decisions and which genes play a key role in the decision. XGBoost provides a way to assess the importance of features. It supports giving each feature a score from 0 to 1. We can determine which features play an important role in decision making and which do not by ranking all features according to their importance scores. This advantage is particularly important for this project because, in addition to wanting to know which prostate cancer secondary site our model is classifying and predicting, we also want to know why this is the case and which genes are most important in the decision.

6.2 Data Balancing

The bar chart on the left in Figure 6.1 shows the distribution of prostate cancer secondary site types in the original dataset, and we note that there is a very unbalanced sample size between the three secondary site types. The unevenness in the number of sample types is likely to have a negative impact on the performance of the model and the fairness of the model being evaluated. As the quality of the dataset is also a key factor in training a good model, we then did a over-sampling data balancing on the original SET_133 dataset.

Firstly, the dataset was randomly divided into 10 sets with the same (or near the same) amount of data and ensured that each set still maintained the same distribution of the three secondary site types. A typical set contains 12 LN, 7 Bone and 4 Liver patient samples. Secondly, within each set, each set is rebalanced by replicating the data for all samples for each secondary site one or more times, so that the number of samples for each site type is approximately equal to the least common multiple of the three site types. Here, we replicate the LN sample data once, Bone twice and Liver five times, at which point there are approximately 24 LNs, 21 Bones and 24 Livers patient samples in a set. When doing 10-fold cross-validation, one set of data at a time is selected as the validation set and the other nine as the training set. This was to ensure that both a sample and its replicate were in the same training or validation set. If a sample appears in the training set and its copy appears in the validation dataset, the validation results will be corrupted.

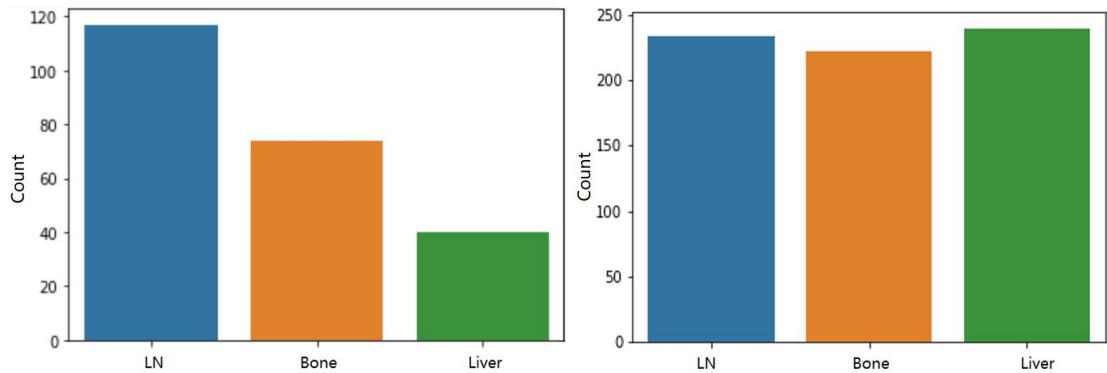


Figure 6.1: Secondary site types distribution in the original dataset (Left) and Over-sampling balanced dataset (right).

6.3 Key Genes Selection

In the first step, we trained the XGBoost model directly on the unbalanced SET_133 dataset and used 10-fold cross-validation to train and validate the model. We obtained a result of 92.21% overall accuracy. In terms of classification accuracy, the model performed well, but we found that not all 133 genes played a role in the predictions. Figure 6.2 shows that only about 40 genes played a significant role and, of these, two were particularly important. However, most genes were not important and even about 40 genes had an importance score of 0, meaning that these genes did not affect prediction at all.

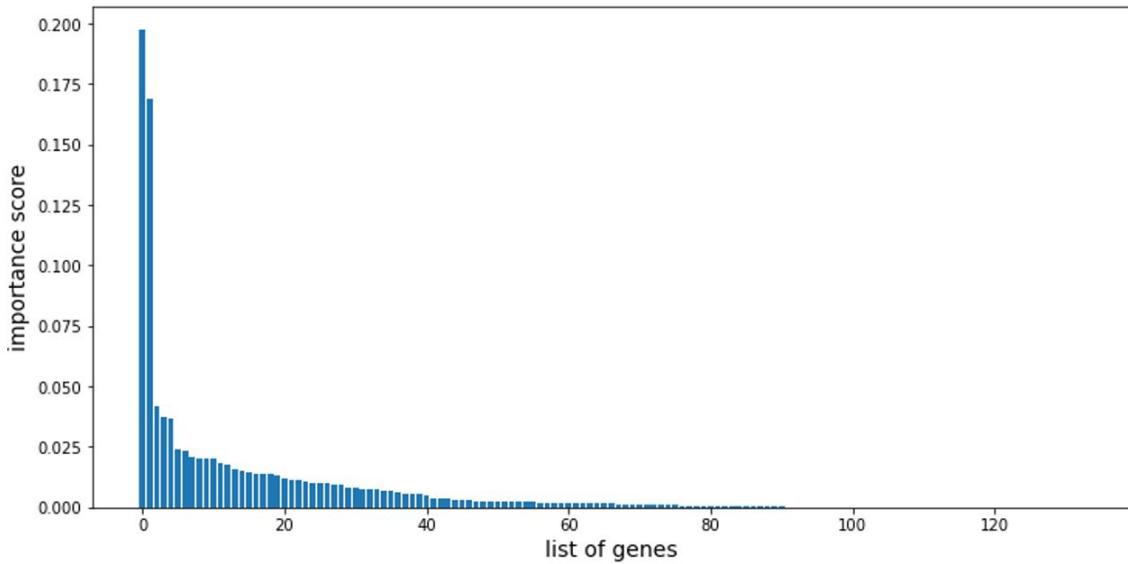


Figure 6.2: Feature importance of all 133 features in the trained XGBoost model.

Based on this observation, we are going to remove the unimportant genes. The

general idea is to train a model, remove the least important genes each time and retrain the model to evaluate its accuracy. By removing the trailing genes step by step and observing the corresponding model accuracy, we find the set of genes used when the accuracy is highest.

We performed the following detailed steps to filter the significant genes.

- a. Train and validate the model by performing a 10-fold cross-validation with the original SET_133 genes. The validation accuracy and feature importance scores in the XGBoost model are then calculated.
- b. Remove the least important genes based on the feature importance scores.
- c. With the remaining genes, train and validate the model using 10-fold cross-validation. Calculate the validation accuracy and feature importance score of the XGBoost model. Repeat 10-fold cross-validation 10 times and take the average accuracy value.
- d. Cycle through steps b to c until only 3 genes remain.
- e. Plot a graph showing the accuracy of model validation versus the number of tail importance genes discarded.
- f. Then apply an oversampling technique to balance the dataset, also doing the above steps.

From the Figure 6.3, we can see that for the original unbalanced SET_133 dataset, the model can still have a high average cross-validation accuracy of 93.58%, even after dropping 99 tail significant genes, i.e. only 34 genes remain. (Orange dots in Figure 6.3.) We named the set of 34 genes SET_34. From the Figure 6.4, we can see that the highest accuracy of 91.97% was obtained with only 13 genes remaining in the case of over-sampling the balanced SET dataset. (Orange dots in the Figure 6.4) We named the set of 13 genes as SET_13.

Comparing these two plots in Figure 6.3 and 6.4, we find that the validation accuracy of the model with the balanced over dataset is generally lower than that of the model with the unbalanced dataset. This does not necessarily mean that the model using balanced data has poorer performance. Instead, as unbalanced data potentially biases the validation result, the model using balanced data may more closely represent the actual performance of the model.

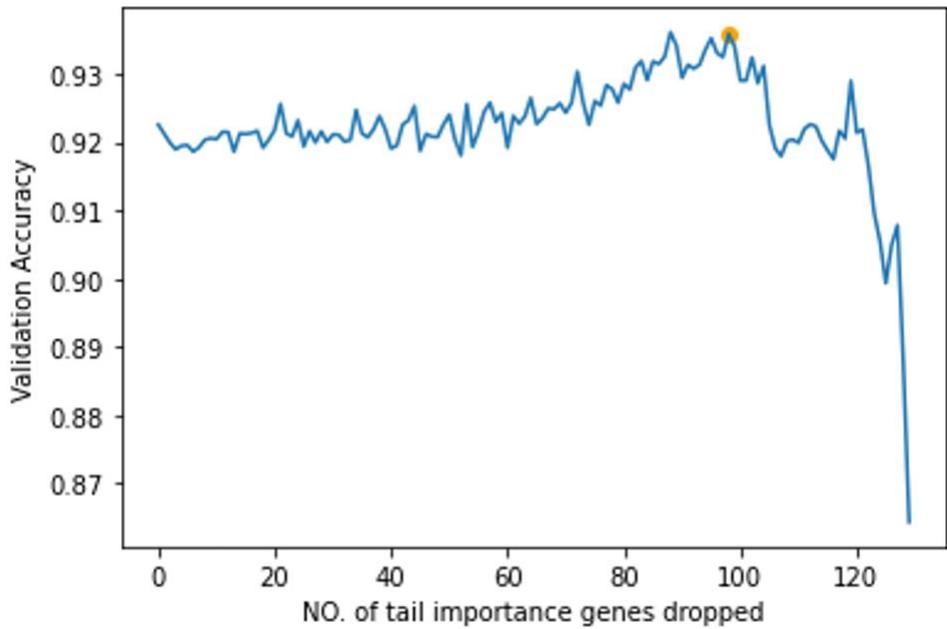


Figure 6.3: *XGBoost model 10-fold cross-validation accuracy against number of tail importance genes dropped with original imbalanced SET_133 dataset.*

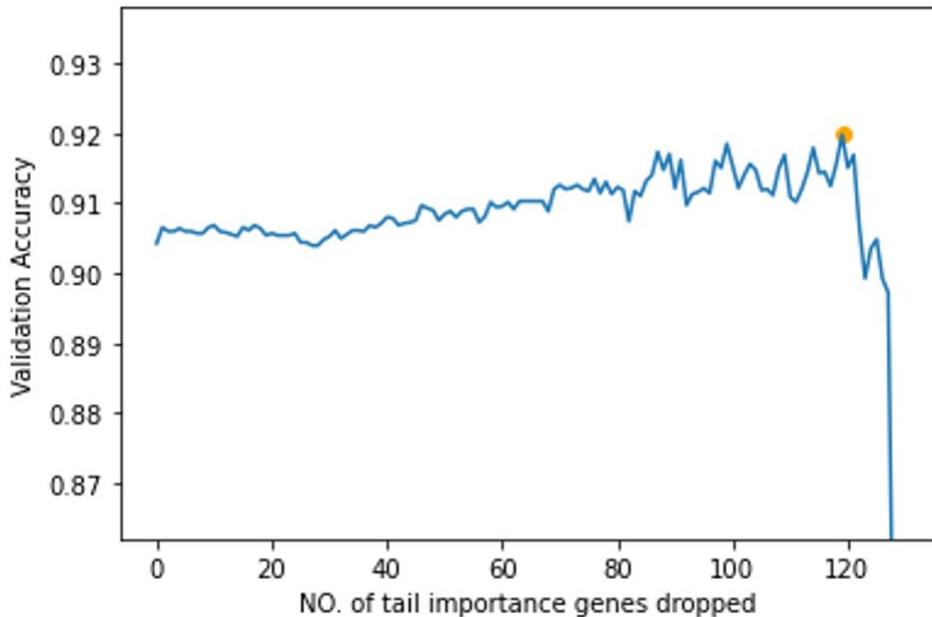


Figure 6.4: *XGBoost model 10-fold cross-validation accuracy against number of tail importance genes dropped with the over-sampling balanced SET dataset.*

Interestingly, the line trend in both Figure 6.3 and 6.4 shows that the validation accuracy in general continues to improve when unimportant genes are eliminated. This contradicts our initial assumption that excluding genes would imply a loss of information. However, this result may suggest that information about prostate cancer metastasis may be encoded in only a very small number of genes.

We also noticed that the accuracy line was a fluctuating increase, rather than a smooth increase when the least important genes were removed each time. This could be explained by the fact that genes may have co-expression patterns, and if we remove a least important co-expressed gene cluster each time, we may get a smoother accuracy graph. However, at present we do not know which genes are in the same co-expression cluster without the further help of our biological knowledge. However, the gene co-expression network analysis may be of some help in this regard.

Another important discovery we made was that for the original dataset, while the best results were obtained when 34 genes were retained, there was also a significant local maximum when 14 genes were retained. Looking further into the details of these genes we found that these the SET_13 genes are all included in these 14 genes, which also implies that the genes in SET_13 are important.

The top important 13 genes in the SET_13 are listed below: PPP1R16A, CXXC1, MAGED2, LRRC66, ERGIC3, GARS, RPL23, RAP1GAP2, IQCE, UBE2L3, UBE2Z, CARD16, PTS.

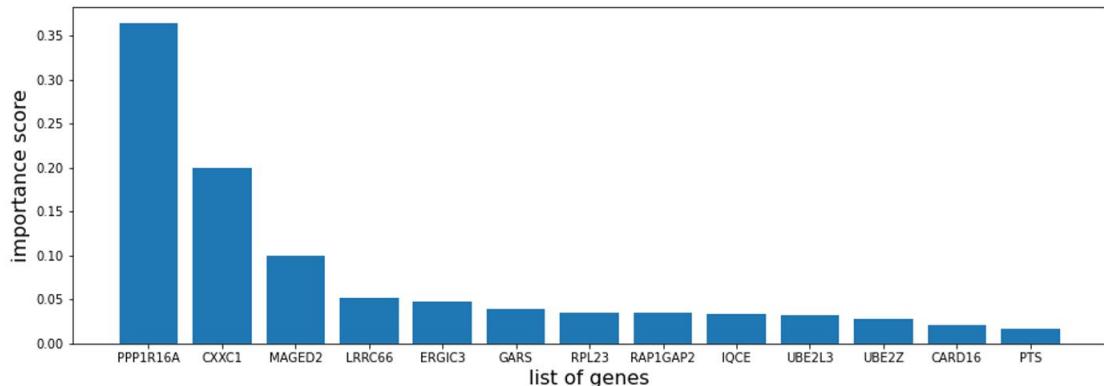


Figure 6.5: *The 13 genes in SET_13 which are considered as very important for prostate cancer metastasis. Genes are ordered in descending order of importance score in the XGBoost classification model.*

6.4 Classification Result Evaluation

The performance of this ML prediction model is impressive, with an overall prediction accuracy of 93.58% and 91.97% on the validation dataset using an unbalanced

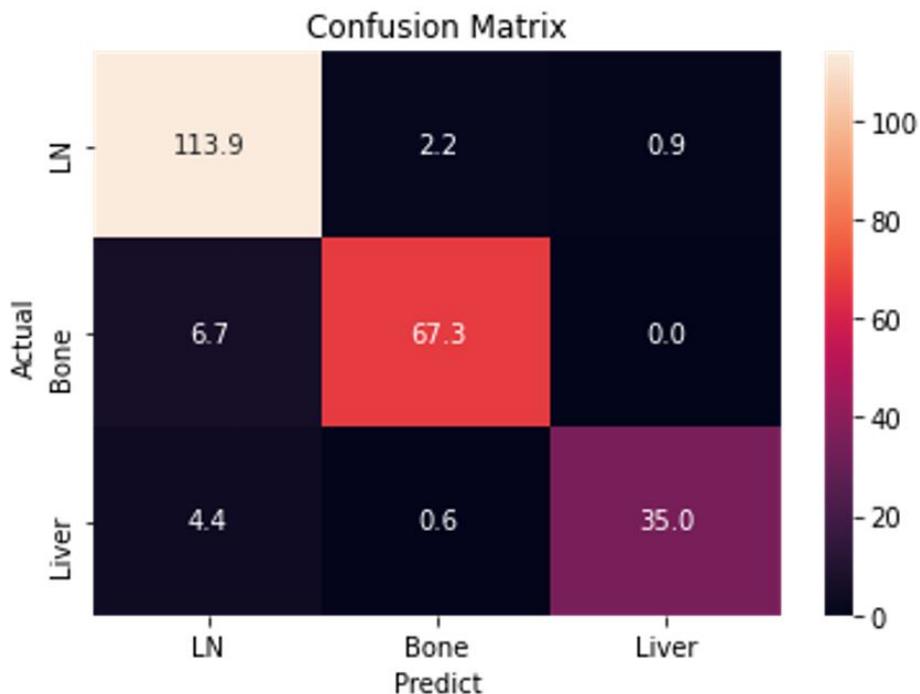


Figure 6.6: *Confusion matrix for the 10-fold cross validation set predictions with original imbalanced dataset.*

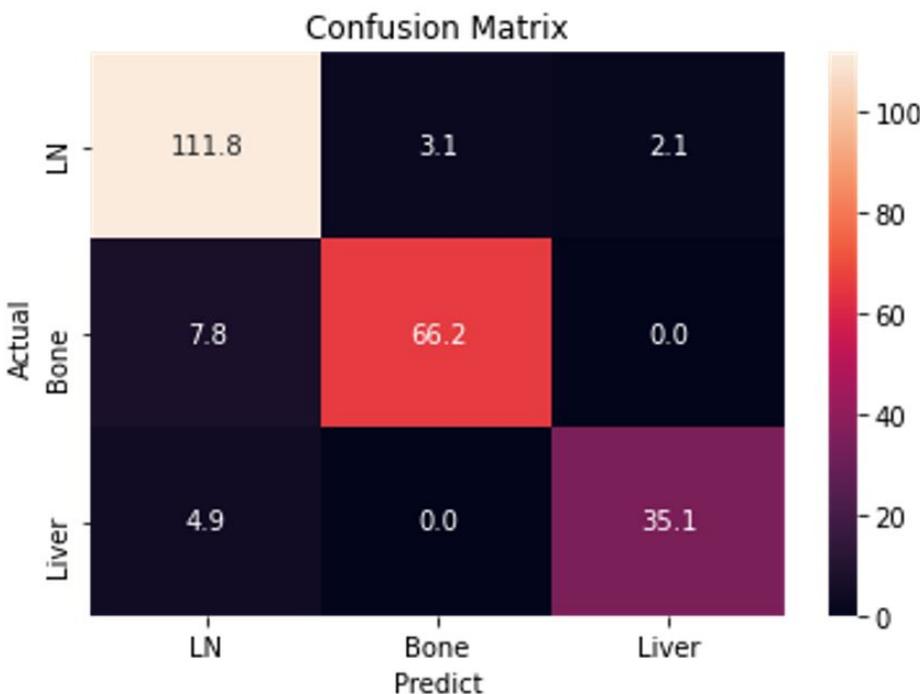


Figure 6.7: *Confusion matrix for the 10-fold cross validation set predictions with the over-sampling balanced dataset.*

dataset and an over-sampled balanced dataset respectively.

As we can see from the confusion matrix and the precision, recall and F1 score ta-

	precision	recall	f1-score
LN	0.91	0.97	0.94
Bone	0.96	0.91	0.93
Liver	0.97	0.88	0.92

Table 6.1: *Precision, recall and F1 score of the XGBoost model prediction result with the original imbalanced dataset.*

	precision	recall	f1-score
LN	0.90	0.96	0.93
Bone	0.96	0.89	0.92
Liver	0.94	0.88	0.91

Table 6.2: *Precision, recall and F1 score of the XGBoost model prediction result with the over-sampling balanced dataset.*

bles, the model can easily classify LN samples correctly, but they also have a slight tendency to classify some bone and liver samples as LN samples. The high precision of these models in classifying bone and liver suggests that few other samples would be misclassified as bone and liver.

These results may indicate that we have selected a very important set of genes for the very accurate classification of secondary sites of prostate cancer. However, the relatively high precision and relatively low recall of the LN classification, and the relatively high precision and the relatively low recall of the bone and liver classification may indicate the following:

- a. The selected genes set may have included most of for LN metastases related genes, but still missed some. This may explain why the model can correctly classify LN samples but also misclassify other samples as LN samples.
- b. The genes set we selected may have contained more than enough bone and liver metastases related genes. And the model may have used these excess genes for LN, bone and liver classification, which resulted in few other samples being misclassified as bone and liver, but some bone and liver samples being misclassified as LN samples.

Chapter 7

Gene Co-expression Network and Community Analysis

7.1 Gene Co-expression Network (GCN)

7.1.1 Intuition of Gene Co-expression Network

As described in the background section, a set of co-expressed genes are involved in the synthesis of a functionally complex protein, and finding a set of genes associated with prostate cancer metastasis would provide a deeper understanding of this cancer. Pure bioanalysis to find co-expressed genes is often difficult and requires a great deal of work. It is fortunate that the way genes are co-expressed is intuitively consistent with the concept of network science. If genes and their co-expression information can be represented as a topological network, advanced network analysis techniques such as community detection can be applied to extract further information and knowledge about genes. Therefore, the concept of a gene co-expression network (GCN) was proposed [Stu03] to represent gene co-expression information and relationships. The construction of a GCN is concept-wise straightforward: nodes represent genes, and if there is an obvious co-expression relationship between a pair of nodes, then an undirected edge is used to connect them.

In this project, we have argued in “Background” section that the Pearson correlation coefficient are often used to measure the relationship between two gene co-expressions. Therefore, gene co-expression networks can be constructed by correlation coefficients between genes.

7.1.2 Dataset

For the GCN analysis, we would like to consider a relatively large gene set, as a small gene set would provide limited information. In this chapter, we will use SET_559 as a start to construct the GCN. In the network analysis, we will also focus on the distribution of SET_133, SET_34 and SET_13 genes in a GCN. We are interested in these gene sets because they are very important gene sets, and any obvious distributive pattern of these critical genes may reveal important underlying knowledge. Note that SET_13 is a subset of SET_34, SET_34 is a subset of SET_133, and SET_133 is a subset of SET_559. We define the set difference notation as follows:

$$A - B = \{x : x \in A \text{ and } x \notin B\} \text{ for set } A \text{ and set } B$$

For example, SET_34 – SET_13 represents a set of genes which in SET_34 but not in SET_13.

7.1.3 Network Construction

We use the 559 nodes in the network to represent each gene in the SET_559. To analyse the GCNs under different prostate cancer secondary site types, we will construct four different GCNs, one for all 231 samples and the other three for each prostate cancer secondary site type. We will refer to the GCN for prostate cancer cells at the LN site as the LN GCN, the GCN for prostate cancer cells at the Bone site as the Bone GCN, the GCN for prostate cancer cells at the Liver site as the Liver GCN, and the GCN for all sample cells as the All-patient GCN. Since the All-patient GCN may contain an ensemble information from all LN, Bone and Liver samples, it is difficult to get clear information by analysing this network, so we just use this network as a reference. Our work will focus on analysing and comparing the LN, Bone and Liver GCNs.

Using the Bone GCN as an example, we build the network by the following steps:

- a. For each of the 559 nodes (genes), add a weighted edge between any two nodes, where the weight value is the absolute value of the correlation coefficient between these two genes. It is important to note that when we

construct the Bone GCN, we should only consider the 74 Bones samples. Therefore, the correlation between the two genes is only calculated based on these 74 values. After this step, we will have a complete weighted network, which means that there is an edge between any two nodes and all the edges are weighted.

- b. Set thresholds from 0.4 to 0.9 with step 0.02. For each threshold, construct the corresponding gene network by removing all edges which weight less than this threshold, and keep all edges which weight greater than or equal to this threshold. Ignore all weights of all edges, then we will obtain an unweighted Bone GCN.
- c. Run non-overlapping community detection algorithm on each gene network, then calculate the network modularity [New06].
- d. Among all the GCNs constructed with candidate thresholds, keep the GCN with the highest modularity value.

The LN and Liver GCNs could also be constructed by the above steps, by considering only the Bone and Liver samples when calculating the correlation values.

As can be seen from the probability distribution function of Pearson correlations for samples at each secondary site in Figure 7.1, most of the edges have low weight values in all three GCNs, which is expected because not most genes are co-expressed with each other. We may be able to construct sparse GCNs, which may make it relatively easy to analyse valuable information. However, we also noted that the LN GCN may differ from the other two GCNs in that there are a relatively large number of genes with very high correlations, suggesting that the LN GCN is denser than the other two GCNs, i.e. the average node degree of the LN may be significantly greater. This could cause difficulties for our later comparisons with these GCNs.

Much of the literature uses weighted GCN for analysis, but co-expression is binary in the sense of biology. This means that two genes are either co-expressed or not co-expressed in the production of a certain protein, with no intermediate situation. For a clear explanation, consider this situation. A gene, 'A', has a correlation of 0.01 with gene 'B', 0.3 with gene 'C', 0.7 with gene 'D' and 0.9 with gene 'E'. It is possible that 'A' has no co-expression relationship with either 'B' or 'C', while it has co-expression relationships with both 'C' and 'D'. Even if the correlation between 'A' and 'C' is significantly greater than the correlation with 'B', it does not necessarily mean that 'A' is more co-expressed with 'C' than with 'B'. Similarly, even if the strong

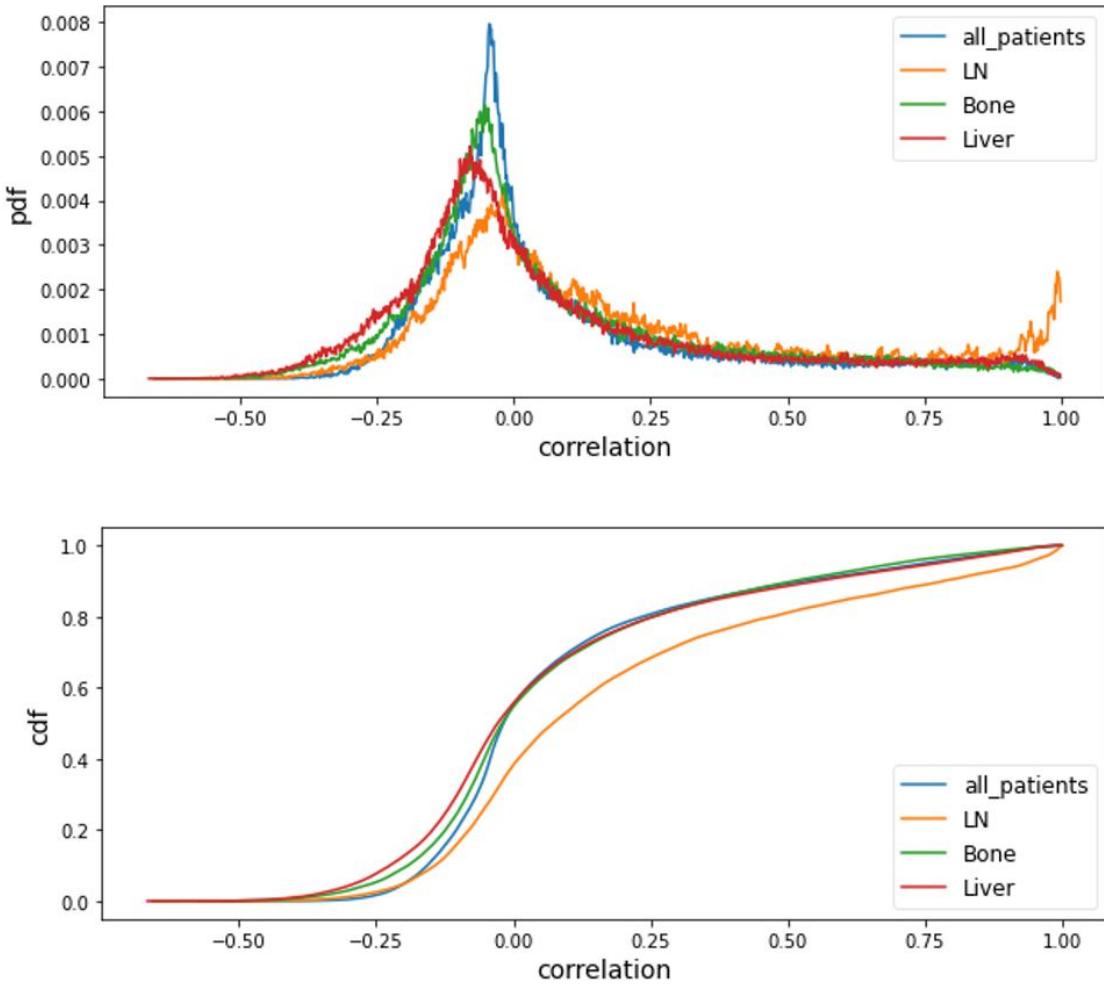


Figure 7.1: *Probability distribution function (PDF) and Cumulative distribution function (CDF) of Pearson correlation between 559 genes for each secondary site patients. Up: PDF, Down: CDF*

relationship between "A" and "D" is significantly smaller than the relationship with "E", it does not necessarily mean that "A" has less co-expression with "D" than with "E". We therefore wish to reduce the impact of this misleading information by setting a threshold and converting the weighted network to unweighted.

7.2 Community Detection

For genes with strong co-expression relationships, strong community patterns are expected in GCN. GCN may have multiple distinct community structures, with each gene community representing a biological cluster of gene co-expressions that are responsible for the synthesis of a number of specific proteins. Therefore, we can perform a community detection on GCN to identify clusters of co-expressed genes. It is likely that some genes are involved in the synthesis of multiple proteins, which

means that there may be overlapping communities in the GCN. However, as this work is an initial study on genetic analysis on metastatic prostate cancer with limited time, we will focus only on non-overlapping community detection to simplify our work.

In network science, a network property called modularity is a measure of the strength of a network's division into communities. Networks with a high degree of modularity tend to have a distinct community structure. Many non-overlapping community detection algorithms have been designed to divide networks into communities by optimising network modularity. The algorithm used in this work is the default non-overlapping community detection algorithm implemented in Gephi, which is also a typical modularity-based algorithm [Vin08].

Before starting the network analysis of the LN, Bone and Liver GCNs, we first performed community detection on the All-patients network to see if there were any obvious communities. The All-patients GCN was constructed from an edge threshold of 0.5, at which point modularity reached a maximum of 0.52. In this network there were 16,704 edges and 502 non-isolated nodes. However, there are 5 tiny groups disconnected from the giant components of the network, so we also treat them as unimportant genes. Thus, there are 492 genes in the network forming 6 distinguishable communities. Since we consider that the GCNs of all patients may contain information about all of LN, Bone and Liver GCNs, we then consider only these 492 genes further in LN, Bone and Liver GCN construction.

In the previous section, we noted that the LN GCN may be somewhat different from the Bone and Liver GCNs. So, we first construct the Bone and Liver GCNs, and then refer to the network characteristics of the Bone and Liver GCNs before constructing the LN GCN. When necessary, we may make some trade-offs on the LN GCN to make the three networks comparable.

7.2.1 Bone GCN

The Bone GCN was constructed from an edge threshold of 0.62, at which point modularity reached a maximum of 0.54. In this network, there are 10,554 edges and 431 nodes in the giant component, which has an average node degree of 48.9. The network is divided into 13 communities and details of the distribution of important genes in these communities are shown in Table 7.1. The visualisation of the Bone GCN can be found in Table 7.5 under section “7.2.4 Community Comparison

Among Different GCNs”, and the ”Colour” column in table 7.1 corresponds to the community colour scheme of the network visualisation in that section. For all genes in each community, the number of genes appearing in SET_13 is placed under the ”SET_13” column, the number of gene appearing in SET_34 but not in SET_13 is placed under ”SET_34 - SET_13”, and so on.

Community		Number of nodes	SET_13	SET_34 - SET_13	SET_133 - SET_34	SET_133 - SET_559
Colour	Ref.					
Violet	1	154	5	8	42	99
Green	2	96	0	1	3	92
Light blue	3	67	5	5	23	34
Orange	4	25	0	1	5	19
Yellow	5	23	2	1	3	17
Red	6	14	1	1	2	10
Deep blue	7	10	0	0	1	9
Gray	8	9	0	0	1	8
Gray	9	8	0	0	4	4
Gray	10	8	0	1	0	7
Gray	11	8	0	0	0	8
Gray	12	7	0	0	1	6
Gray	13	2	0	0	1	1
Total		431 (Giant component)	13	18	86	314

Table 7.1: *Bone GCN community details*

7.2.2 Liver GCN

The Liver GCN was constructed from an edge threshold of 0.54, at which point modularity reached a maximum of 0.33. In this network, there are 15,748 edges and 477 nodes in the giant component, which has an average node degree of 66.4. The network is divided into 6 communities and details of the distribution of important genes in these communities are shown in Table 7.2.

7.2.3 LN GCN

By the GCN construction procedure in section “7.1.3 Network Construction”, the LN GCN was constructed from an edge threshold of 0.24, at which point modularity reached a maximum of 0.19. In this network, there are 50,477 edges and 492 nodes in the giant component, which has an average node degree of 205.2. This network is very different from the Bone and Liver GCN in terms of network size, average node degree and modularity. The low threshold allows a large number of edges to

Community		Number of nodes	SET_13	SET_34	SET_133	SET_133
Colour	Ref.			- SET_13	- SET_34	- SET_559
Violet	1	171	4	3	21	143
Green	2	164	8	8	33	115
Light blue	3	124	1	7	33	83
Orange	4	25	0	0	3	22
Yellow	5	2	0	0	1	1
Red	6	2	0	0	1	1
Total		474 (Giant component)	13	18	94	349

Table 7.2: Liver GCN community details

Community		Number of nodes	SET_13	SET_34	SET_133	SET_133
Colour	Ref.			- SET_13	- SET_34	- SET_559
violet	1	212	5	6	41	160
green	2	115	3	6	26	80
Light blue	3	40	3	3	6	28
Orange	4	25	0	0	3	22
Yellow	5	10	0	0	1	9
Red	6	9	1	1	0	7
Deep Blue	7	8	0	0	1	7
Pink	8	5	0	0	0	5
Total		424 (giant component)	12	16	78	318

Table 7.3: LN GCN community details

be preserved, making the network very dense. We also noted that if we use larger thresholds, the number of nodes in the giant component becomes smaller, but the average node degree of the network remains high and the modularity of the network keeps decreasing. In order to make the LN GCN comparable to the Bone and Liver GCNs, we made a trade-off here by customising the threshold so that the size of the giant component of the LN GCN is similar to that of the Bone and Liver GCNs. The LN GCN was then constructed from an edge threshold of 0.56, with 25,685 edges, 424 nodes in the giant component, which has an average node degree of 118.9, eight communities and a modularity of 0.10.

7.2.4 Community Comparison Among Different GCNs

We made a summarization of the All-patient GCN, LN GCN, Bone GCN and Liver GCN and community details in table 7.4. To be noticed that, since we are interested in the distribution of the SET_13 genes among the communities, we assign index

Gene Co-expression Networks	All-patients GCN	LN GCN	Bone GCN	Liver GCN
Number of samples	231	117	74	40
Number of genes under study	559	492	492	492
Correlation Threshold	0.50	0.56	0.62	0.54
Number of non-isolated nodes	502	432	443	477
Number of links	16,704	25,685	10,554	15,748
Average node degree	66.5	118.9	47.6	66.0
Modularity	0.52	0.10	0.54	0.33
Giant Component				
Giant Component size	492	424	431	474
Number of links	16,699	25,681	10,548	15,746
Average node degree	67.9	121.1	48.9	66.4
Modularity	0.52	0.10	0.54	0.33
Community				
Number of communities	6	8	13	6
1st community size (list of key genes)	154 (2,3,6)	212 (2,6,8,11,12)	154 (0,2,3,11,12)	171 (2,3,6,11)
2nd community size (list of key genes)	130 (0,11,12)	115 (4,5,9)	96	164 (0,1,5,7,8,9,10,12)
3rd community size (list of key genes)	70 (7,10)	40 (0,7,10)	67 (1,5,6,8,10)	124 (4)
4th community size (list of key genes)	60	25	25	25
5th community size (list of key genes)	51 (1,5,8,9)	10	23 (7,9)	2
6th community size (list of key genes)	27 (4)	9 (1)	14 (4)	2
(isolated key genes)		(3)		

Table 7.4: *Summary of the All-patient, LN, Bone and Liver GCN communities.*

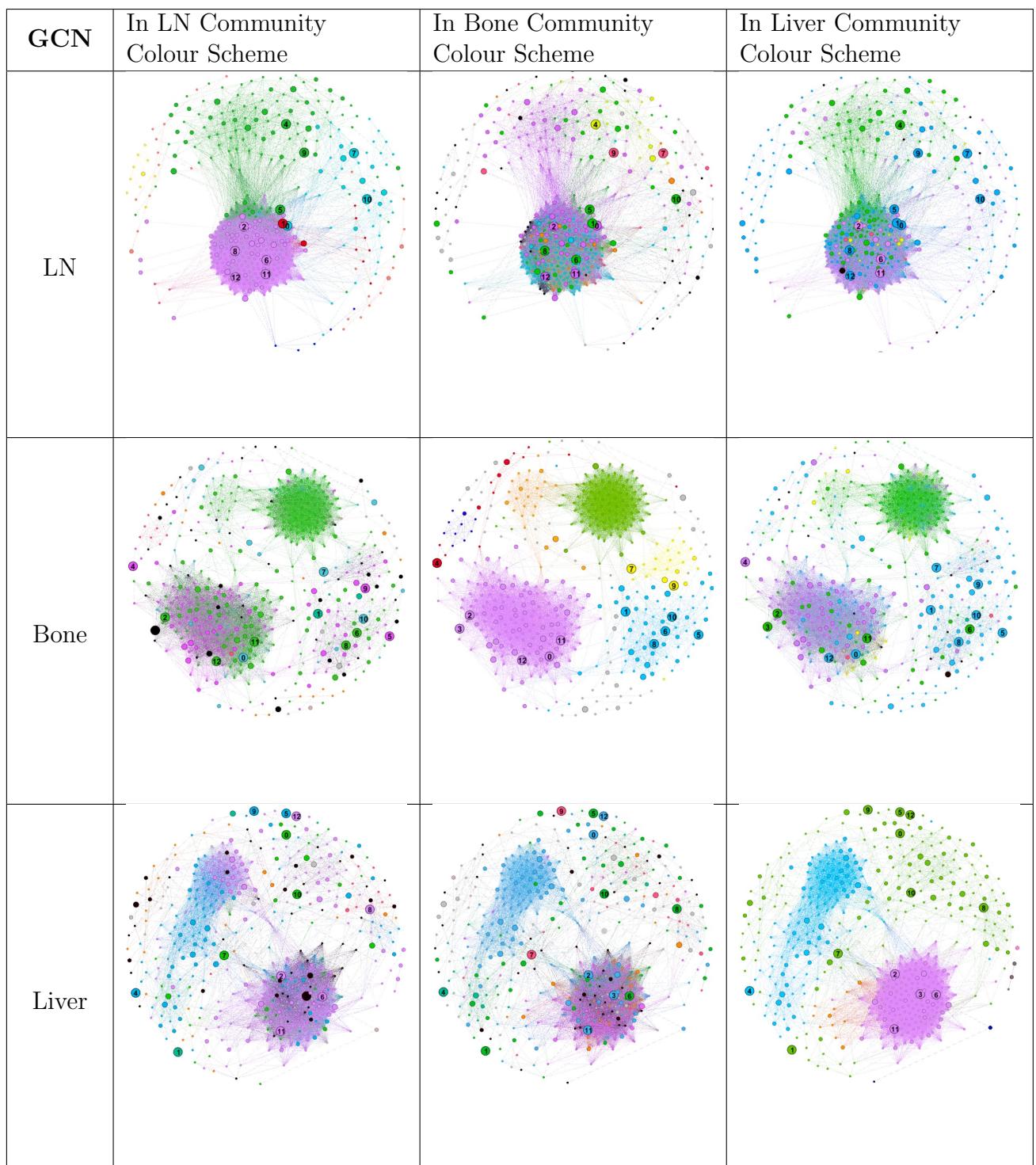


Table 7.5: *LN, Bone and Liver GCNs visualization in different GCN community colour schemes.*

numbers from 0 to 12 for each of the 13 key genes. In table 7.4, we also provided a list of the key genes' indexes in each community in round bracket.

To compare the gene communities in different prostate cancer secondary site GCNs, we applied LN, Bone and Liver GCN community colour schemes for each of the GCNs. For the graphs in table 7.5, each row is community visualisations for one prostate cancer secondary site GCN in different GCNs' community colour schemes. The layout of GCNs in same row keep the same, but genes may belong to different communities. However, each column is community visualisations for each GCN in one GCN's community colour schemes. The graphs in the diagonal represent the GCNs in itself community colour scheme. We also use different node size to denote the importance of genes. The most important genes are SET_13 genes, which are in the largest node size, and also annotated the index number of the 13 genes on each node. The second largest nodes are genes in SET_34 but not in SET_13, and the third largest nodes are genes in SET_133 but not in SET_34. The smallest nodes are genes in SET_559 but not in SET_133. A larger view of these GCN visualisations are also added in Appendix.

7.3 Evaluation

We can find there are strong gene community structures in these GCNs, particularly in the bone and liver GCNs. By altering community colour schemes with each GCN, we can also see that gene communities that occur in one GCN often occur in another GCN, which is strong evidence that gene community members tend to co-express.

However, we also noted that some genes went into different gene communities in different GCNs and there was some variation in the members that made up each community. This may be because the same genes play different roles in different kinds of metastatic cancer cells. In other words, different types of prostate cancer cells have different co-expression community structures, which allow the cells to synthesise different proteins and thus determine differences in cellular properties.

By deeply analysing the distribution of SET_13 genes in the communities of different GCNs, we found that the key genes tended to be in different gene communities in different GCNs. For example, SET_13 key genes with indexes 4, 5 and 9 were in the same community in the LN GCN but in three different communities in the Bone GCN. While in the Liver GCN, key genes 5 and 9 were in the same community and

4 was in another community. This apparent difference in the distribution of key genes could explain the effectiveness of the machine learning model.

Chapter 8

Discussion

8.1 Achievements

In this interdisciplinary study, we have used data science, machine learning and network science methods to perform genetic analysis of metastatic prostate cancer and achieved promising results. In Chapter 5, we perform a thorough analysis of the raw dataset and then propose a general approach to pre-process the gene expression dataset. We filtered out 2 important gene sets, SET_559 and SET_133, using only the data analysis method. In Chapter 6, we further extracted two key gene sets, SET_34 and SET_13, from SET_133 using the XGBoost model and trained a high-performance classification model using these genes. In Chapter 7, with intuition from biology and network science, we constructed gene co-expression networks for LN, Bone and Liver tumour samples respectively, and then performed network analysis and community detection on these networks. The results show that prostate cancer related genes have a strong gene community structure in a variety of metastatic cancer cells, and that these gene communities may play different roles, leading to cancer metastasis to different secondary sites.

8.2 Methodological Contribution

8.2.1 Originality

(i) XGBoost model

In much of the literature on cancer cell classification and prediction based on sample gene expression values, machine learning models such as neural networks or

support vector machine models are used. While these powerful models can produce accurate results by discovering underlying patterns behind the data, it suffers from interpretability problems. In this work, we used a simple decision tree-based model called XGBoost, as it supports scoring the importance of features. With the help of the feature importance scoring function of the XGBoost model, we successfully extracted several very important genes and achieved a high classification performance.

(ii) Network analysis

Another unique aspect of this work is that we constructed multiple GCNs, each for one type of prostate cancer secondary site. Most literature analyses only one GCN, and a single network may provide only limited or unclear information. In this work, however, we compare and contrast the LN, Bone and Liver GCNs and attempt to discover new knowledge and patterns from them. We compare these GCNs by their network characteristics, network layout and community structure, and use visualisation to highlight these differences in a more engaging way. We also delved into the differences in gene communities across GCNs by tracking key genes.

(ii) Ensemble methodology

While much of the literature has been devoted to the genetic analysis of cancer using pure analysis methods, machine learning methods and gene co-expression network analysis methods respectively, very little work has combined them together to solve a complex problem. The most prominent approach in our work is to combine these three methods by taking advantage of their strengths.

In the preliminary work, we did a solid data analysis of the original dataset, and this process gave us a concrete understanding of the dataset, which will in a way make our next work easier and more sensible. We then used data analysis methods to pre-filter a number of genes to reduce the number of features used in the machine learning model, thereby improving the performance of the machine learning methods. Then, although with the help of powerful machine learning methods, we further extracted key genes and were able to accurately classify different secondary cells, we still did not know how genes were co-expressed at different metastatic sites. Therefore, we constructed GCNs for network analysis to reveal the co-expression patterns of genes. Also, with the help of machine learning results, we could track the different

roles of key genes in different GCNs.

Through this research experience, we have gained a deeper understanding that each method has its advantages and disadvantages. It is very hard to solve a complex problem using only one method, but we can use the advantages of multiple methods to discover new insights to solve the problem. These methods are like rings in a chain, and the trickiest part is not how to use individual methods, but how to use the strengths of one method to complement the weaknesses of another, and find a way to bring these methods together to solve a complex problem.

8.2.2 Transferability

The methods used in this project are transferable, which means that the genetic analysis methods covered in this paper can also be applied to other gene expression datasets, and even other cancer research tasks. Although the current study was aimed at metastatic prostate cancer, none of our analytical methods were designed specifically for metastatic prostate cancer. Only the basic biology associated with cancer was considered in the methodological design.

8.3 Limitation

The pre-processed dataset used in this project contains only 231 samples from 3 types of patients. Also the sample size was very unevenly distributed between these 3 types. The small sample size and the few types of metastatic prostate cancer secondary sites may make the results of this study less generalisable. We therefore found another dataset of gene expression values for metastatic prostate cancer. However, due to the limited time available for this study, we did not validate the performance and compatibility of our trained XGBoost model, and the consistency of our GCNs on this dataset.

In the GCN analysis section, we only used the non-overlapping community detection algorithm to discover co-expressed genes. However, in terms of biology, some genes may be involved in the synthesis of multiple proteins, which means that there may be overlapping communities in the GCN. Therefore, it makes more sense to use an overlapping community detection algorithm instead. However, there are many overlapping community detection algorithms available, each may be suitable for a different application scenario. It would take a lot of time to investigate which al-

gorithm is appropriate, and this may be beyond the scope of this project. There is also the fact that the analysis of overlapping communities is much more complex. As this work is a preliminary study of the genetic analysis of prostate cancer and time is limited, we have made a trade-off here by focusing only on detection of non-overlapping communities to simplify the problem.

Chapter 9

Future Work

9.1 ML Models

Many publications mention that SVM models may be a good choice for tasks such as cancer classification. SVM with recursive feature selection can also be effective for selecting key genes. Therefore, in the next work, other feasible machine learning model solutions including SVM can be implemented and it would be interesting to compare these models together with XGBoost models and conclude which one works best. And, with a new dataset of gene expression values for metastatic prostate cancer recently collected by the Institute of Cancer Research in London, we can next train these machine models on the existing dataset used in this study, and then validate all of our trained models on this new dataset in order to verify which one is the best in terms of compatibility and performance.

9.2 Gene Co-expression Network

As we can see in the GCN visualisations, many communities have a large size, which could contain too many genes may not necessarily have co-expression relationship. So, in next step, we could do sub-community detection on large communities to further break the GCN into relatively small pieces. And then compare the gene communities in different GCNs, we may spot communities in different GCNs more consistent.

In addition, it is of considerable significance to do overlapping community detection in future work, as overlapping community detection is more closely related to the significance of gene co-expression in biology. Furthermore, this contribution will be of great help in future studies to develop drugs related to prostate cancer metastasis.

For genes that are in overlapping regions of multiple communities, the drugs made may need to avoid affecting these genes as much as possible. This is because the presence of these genes in multiple co-expressed gene communities implies that if a drug has an effect on such genes, it may affect the cell's synthesis of multiple functional proteins, producing some complex side effects.

Bibliography

- [cBioPortal] cBioPortal. Cancer Genomics dataset website. Official website available at: . Accessed on 23/04/2021
- [HUGO] Human Genome Organisation. Official website available at: . Accessed on 23/04/2021
- [WHO14] World Cancer Report. World Health Organization World Cancer Report Chapter 1.1. 2014. ISBN 978-9283204299.
- [Fie16] Fiedler K, Lazzaro S, Lutz J, Rauch S, Heidenreich R. "mRNA Cancer Vaccines". Recent Results in Cancer Research. Fortschritte der Krebsforschung. Progres dans les Recherches Sur le Cancer. 2016). 209: 61–85. doi:10.1007/978-3-319-42934-2_5. ISBN 978-3-319-42932-8. PMID 28101688.
- [Ire16] Irengbam Rocky Mangangcha, Md. Zubbair Malik, Omer Kucuk, Shakir Ali, R.K. Brojen Singh. Kinless hubs are potential target genes in prostate cancer network, Genomics, Volume 112, Issue 6. 2020, Pages 5227-5239, ISSN 0888-7543,
- [Liu16] Liu SJ, Nowakowski TJ, Pollen AA, Lui JH, Horlbeck MA, Attenello FJ, He D, Weissman JS, Kriegstein AR, Diaz AA, Lim DA (April 2016). "Single-cell analysis of long non-coding RNAs in the developing human neocortex". Genome Biol. 17: 67. doi:10.1186/s13059-016-0932-1. PMC 4831157. PMID 27081004.
- [Bio14] Brooker RJ, Widmaier EP, Graham LE, Stiling PD (2014). Biology (Third international student ed.). New York, NY: McGraw Hill Education. p. 249. ISBN 978-981-4581-85-1.
- [RNA-Seq] Chu Y, Corey DR (August 2012). "RNA sequencing: platform selection, experimental design, and data interpretation". Nucleic Acid Therapeutics. 22 (4): 271–4. doi:10.1089/nat.2012.0367. PMC 3426205. PMID 22830413.

- [Cor95] Cortes, Corinna; Vapnik, Vladimir N. (1995). "Support-vector networks" (PDF). *Machine Learning*. 20 (3): 273–297. CiteSeerX 10.1.1.15.9362. doi:10.1007/BF00994018. S2CID 206787478.
- [XGBPy] XGBoost Python Package project GitHub website. <https://github.com/dmlc/xgboost>
- [Bel57] Bellman, Richard Ernest; Rand Corporation (1957). Dynamic programming. Princeton University Press. p. ix. ISBN 978-0-691-07951-6.,
- [Tet95] Tetko, I. V.; Livingstone, D. J.; Luik, A. I. (1995). "Neural network studies. 1. Comparison of Overfitting and Overtraining" (PDF). *Journal of Chemical Information and Modeling*. 35 (5): 826–833. doi:10.1021/ci00027a006.
- [Soo11] Soohyun Lee, Chae Hwa Seo, Byungho Lim, Jin Ok Yang, Jeongsu Oh, Minjin Kim, Sooncheol Lee, Byungwook Lee, Changwon Kang, Sanghyuk Lee, Accurate quantification of transcriptome from RNA-Seq data by effective length normalization, *Nucleic Acids Research*, Volume 39, Issue 2, 1 January 2011, Page e9
- [Soh17] Soh, K., Szczurek, E., Sakoparnig, T. et al. Predicting cancer type from tumour DNA signatures. *Genome Med* 9, 104 (2017). <https://doi.org/10.1186/s13073-017-0493-2>
- [Car06] Caruana R, Niculescu-Mizil A. An empirical comparison of supervised learning algorithms. In: Proceedings of the 23rd International Conference on Machine Learning. ICML '06. New York: ACM; 2006. p. 161–8.
- [Vog04] Vogelstein B, Kinzler KW. Cancer genes and the pathways they control. *Nat Med*. 2004;10(8):789–99.
- [Vog13] Vogelstein B, Papadopoulos N, Velculescu VE, Zhou S, Diaz LA, Kinzler KW. Cancer genome landscapes. *Science*. 2013;339(6127):1546–58.
- [Gol99] Golub T, Slonim D, Tamayo P, Huard C, Gaasenbeek M, Mesirov J, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*. 1999;286(5439):531–7.
- [PCC] SPSS TUTORIALS: PEARSON CORRELATION. Available at: <https://libguides.library.kent.edu/SPSS/PearsonCorr>

- [Kha01] Khan J, Wei J, Ringnér M, Saal L, Ladanyi M, Westermann F, et al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nat Med.* 2001;7(6):673–9.
- [Ramaswamy S, Tamayo P, Rifkin R, Mukherjee S, Yeang C, Angelo M, et al. Multiclass cancer diagnosis using tumor gene expression signatures. *Proc Natl Acad Sci USA.* 2001;98(26):15149–54.
- [Tib02] Tibshirani R, Hastie T, Narasimhan B, Chu G. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proc Natl Acad Sci USA.* 2002;99(10):6567–72.
- [Kan17] Kang S, Li Q, Chen Q, Zhou Y, Park S, Lee G, et al. CancerLocator: non-invasive cancer diagnosis and tissue-of-origin prediction using methylation profiles of cell-free DNA. *Genome Biol.* 2017;18(1):53.
- [Hao17] Hao X, Luo H, Krawczyk M, Wei W, Wang W, Wang J, et al. DNA methylation markers for diagnosis and prognosis of common cancers. *Proc Natl Acad Sci USA.* 2017;114(28):7414–9.
- [Can15] Cantini, L., Medico, E., Fortunato, S. et al. Detection of gene communities in multi-networks reveals cancer drivers. *Sci Rep* 5, 17386 (2015). <https://doi.org/10.1038/srep17386>
- [XGB16] Chen, Tianqi; Guestrin, Carlos (2016). "XGBoost: A Scalable Tree Boosting System". In Krishnapuram, Balaji; Shah, Mohak; Smola, Alexander J.; Aggarwal, Charu C.; Shen, Dou; Rastogi, Rajeev (eds.). Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016. ACM. pp. 785–794. arXiv:1603.02754. doi:10.1145/2939672.2939785.
- [New06] Newman, M. E. J. (2006). "Modularity and community structure in networks". *Proceedings of the National Academy of Sciences of the United States of America.* 103 (23): 8577–8696. arXiv:physics/0602124. Bibcode:2006PNAS..103.8577N. doi:10.1073/pnas.0601602103. PMC 1482622. PMID 16723398.
- [Bra18] Bray F, Ferlay J, Soerjomataram I, Siegel RL, Torre LA, Jemal A (November 2018). "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries". *CA: A Cancer Journal for Clinicians.* 68 (6): 394–424. doi:10.3322/caac.21492. PMID 30207593. S2CID 52188256

- [NCI14] "Prostate Cancer Treatment (PDQ) – Health Professional Version". National Cancer Institute. 2014-04-11. Archived from the original on 5 July 2014. Retrieved 1 July 2014.
- [Rud15] Ruddon RW (2007). Cancer biology (4th ed.). Oxford: Oxford University Press. p. 223. ISBN 978-0195175431. Archived from the original on 2015-09-15.
- [Wag12] Wagner, G.P., Kin, K. Lynch, V.J. Measurement of mRNA abundance using RNA-seq data: RPKM measure is inconsistent among samples. *Theory Biosci.* 131, 281–285 (2012). <https://doi.org/10.1007/s12064-012-0162-3>
- [Zha05] Zhang, Bin and Horvath, Steve (2005) "A General Framework for Weighted Gene Co-Expression Network Analysis," *Statistical Applications in Genetics and Molecular Biology*: Vol.4: Iss. 1, Article 17. DOI: 10.2202/1544-6115.1128
- [Sav20] Savary, C., Kim, A., Lespagnol, A. et al. Depicting the genetic architecture of pediatric cancers through an integrative gene network approach. *Sci Rep* 10, 1224 (2020). <https://doi.org/10.1038/s41598-020-58179-0>
- [Stu03] Stuart, Joshua M; Segal, Eran; Koller, Daphne; Kim, Stuart K (2003). "A gene coexpression network for global discovery of conserved genetic modules". *Science*. 302 (5643): 249–55. Bibcode:2003Sci...302..249S. CiteSeerX 10.1.1.119.6331. doi:10.1126/science.1087447. PMID 12934013. S2CID 3131371.
- [Vin08] Gephi modularity algorithm. Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, Etienne Lefebvre, Fast unfolding of communities in large networks, in *Journal of Statistical Mechanics: Theory and Experiment* 2008 (10), P1000
- [Jap98] Japaridze, G., and De Jongh, D. (1998) "The logic of provability" in Buss, S., ed., *Handbook of Proof Theory*. North-Holland: 476–546.

Appendix A

Model hyperparameters

Hyperparameters	Value	Hyperparameters	Value
base_score	0.5	booster	gbtree
colsample_bylevel	1	colsample_bynode	1
colsample_bytree	1	seed	0
learning_rate	0.1	gamma	0
max_depth	3	max_delta_step	0
missing	None	min_child_weight	1
nthread	1	n_estimators	100
reg_alpha	0	objective	binary:logistic
scale_pos_weight	1	reg_lambda	1
subsample	1	verbosity	1

Table A.1: XGBoost model Hyperparameters setting

Appendix B

Figures

B.1 Effective of mask selection method

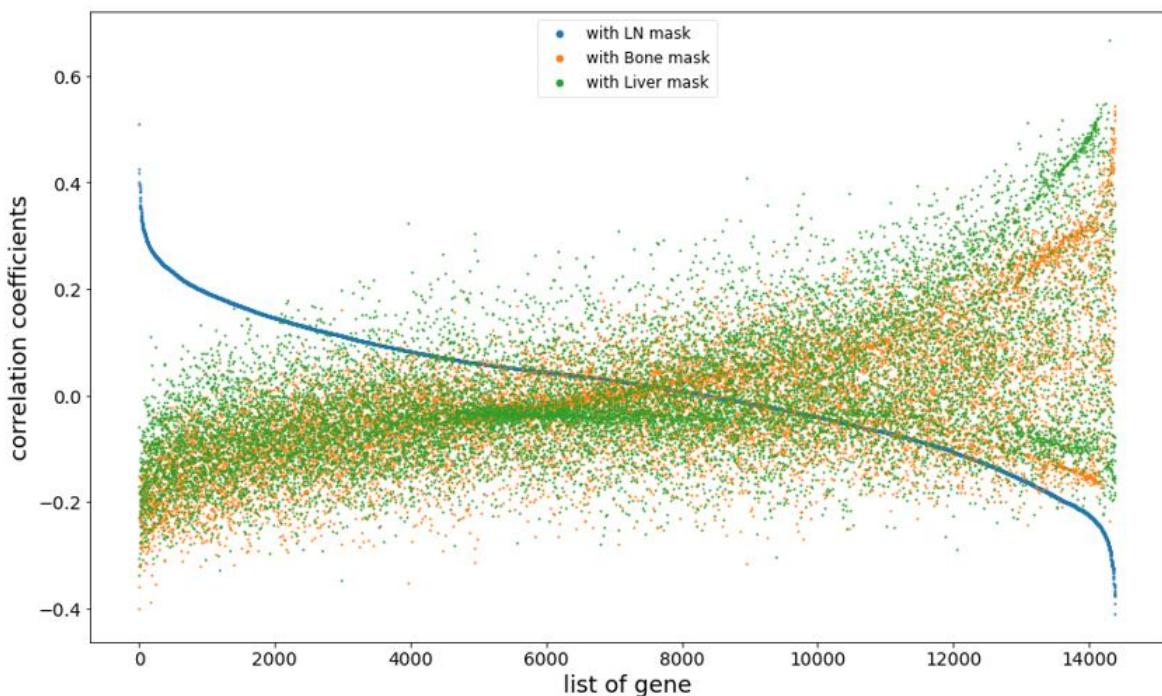


Figure B.1: X are 14,379 genes and Y are 3 correlation values with each of the LN,Bone and Liver metastases masks in different colours. Sort the genes in descending order according to the correlation value with the LN metastases mask.

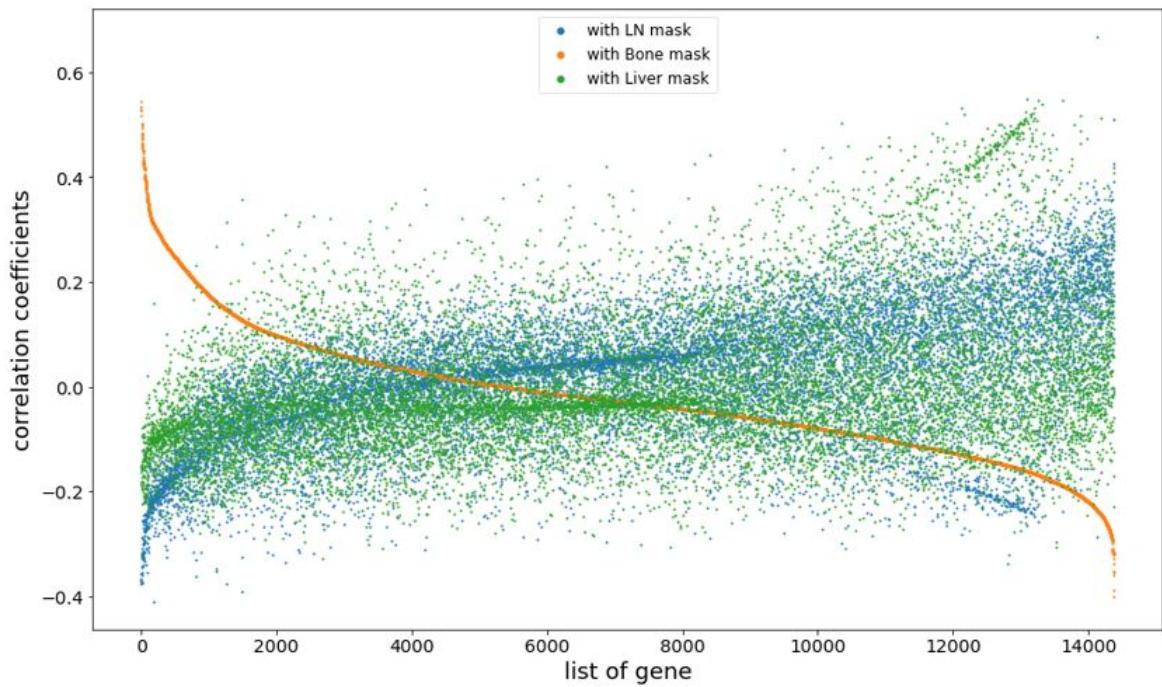


Figure B.2: X are 14,379 genes and Y are 3 correlation values with each of the LN,Bone and Liver metastases masks in different colours. Sort the genes in descending order according to the correlation value with the LN metastases mask.

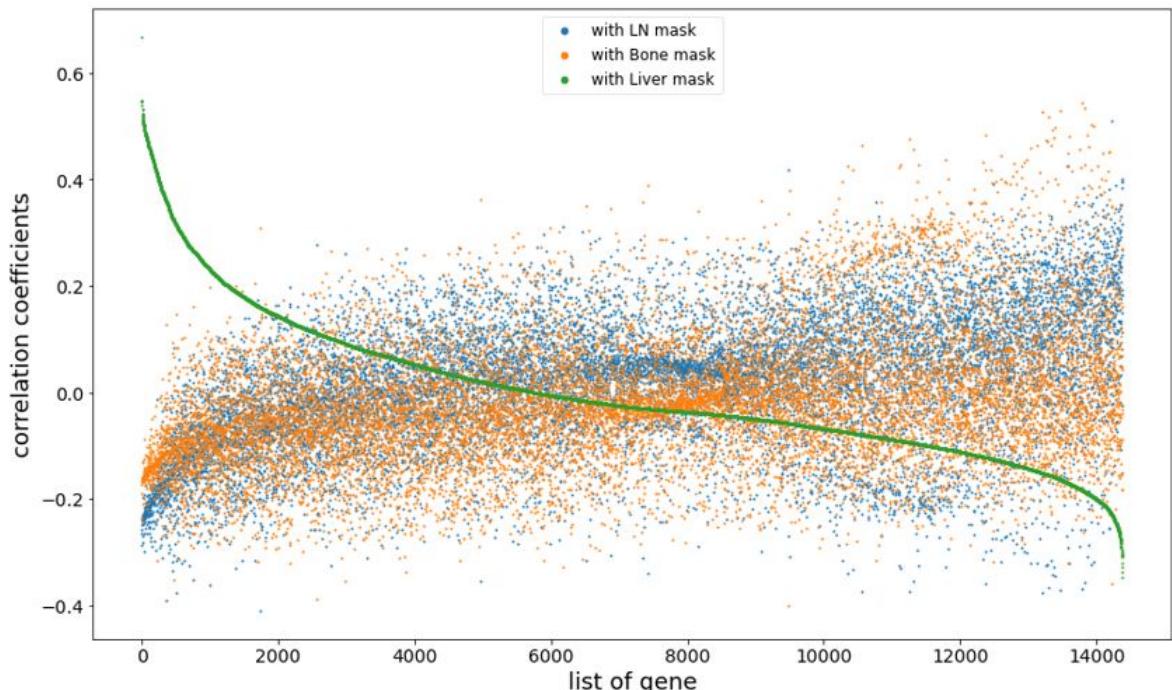


Figure B.3: X are 14,379 genes and Y are 3 correlation values with each of the LN,Bone and Liver metastases masks in different colours. Sort the genes in descending order according to the correlation value with the Liver metastases mask.

B.2 Gene Co-expression Networks Visualisation

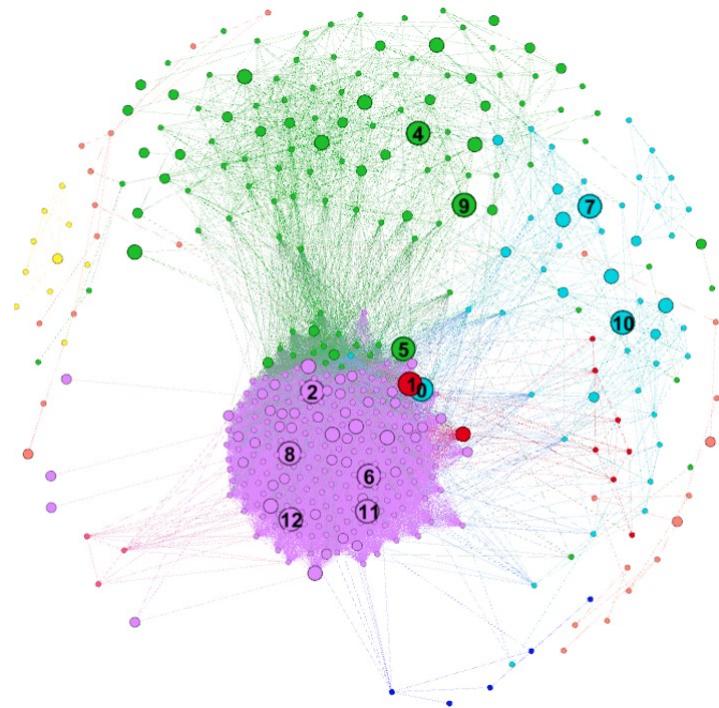


Figure B.4: LN GCN

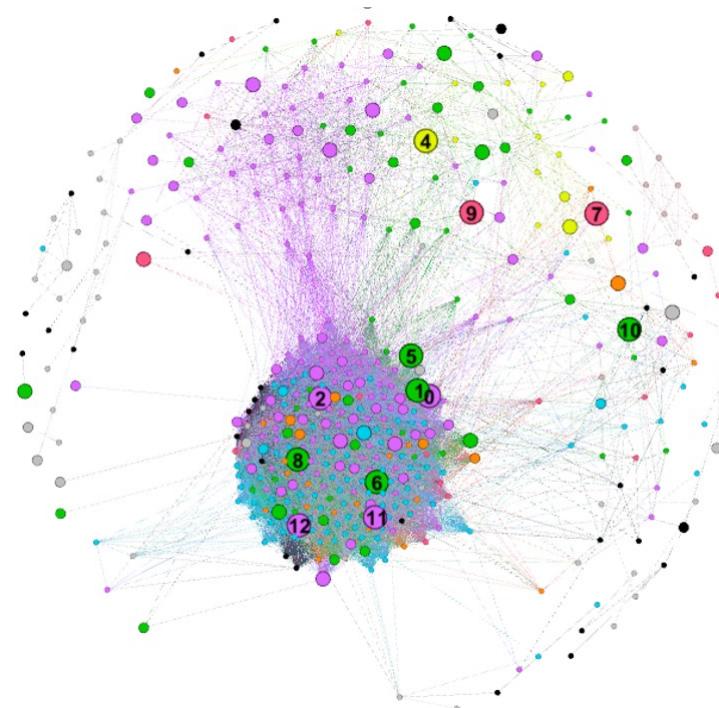


Figure B.5: LN GCN coloured by Bone GCN's community colour scheme

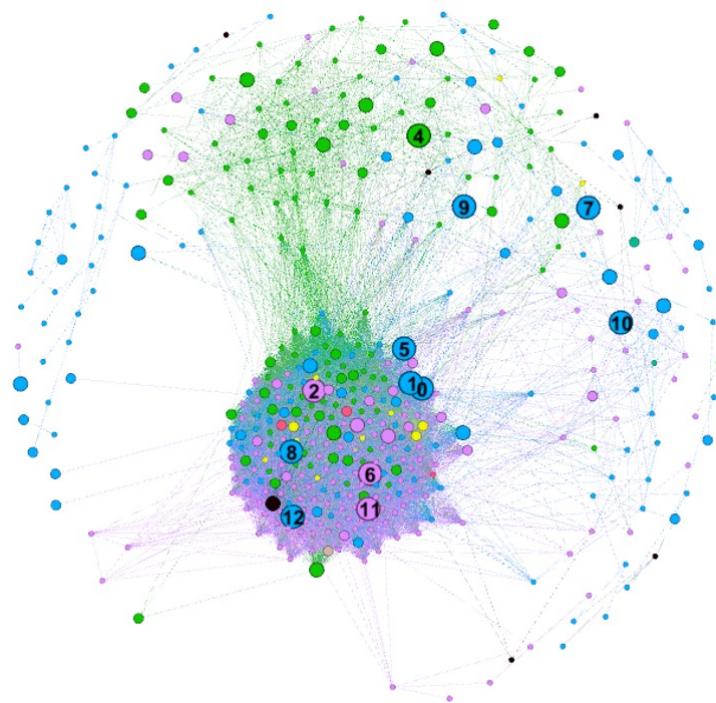


Figure B.6: LN GCN coloured by Liver GCN's community colour scheme

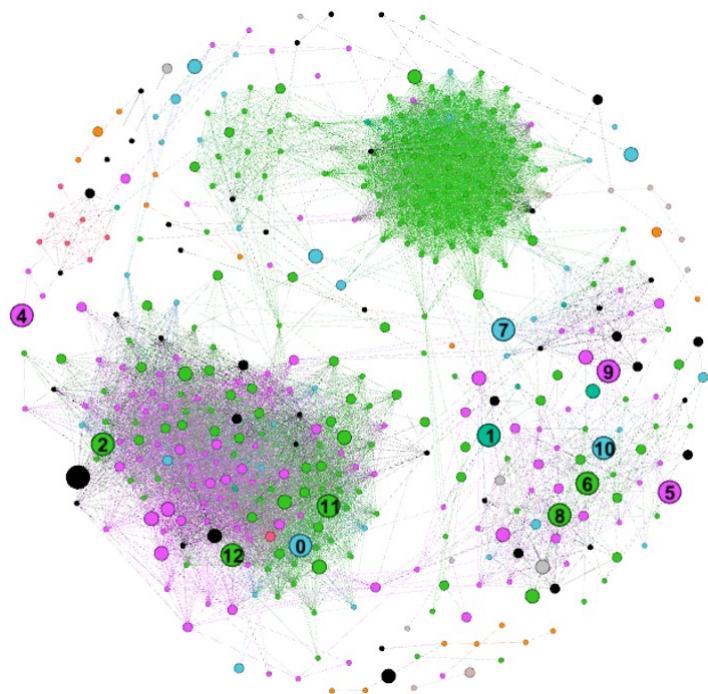


Figure B.7: Bone GCN coloured by Ln GCN's community colour scheme

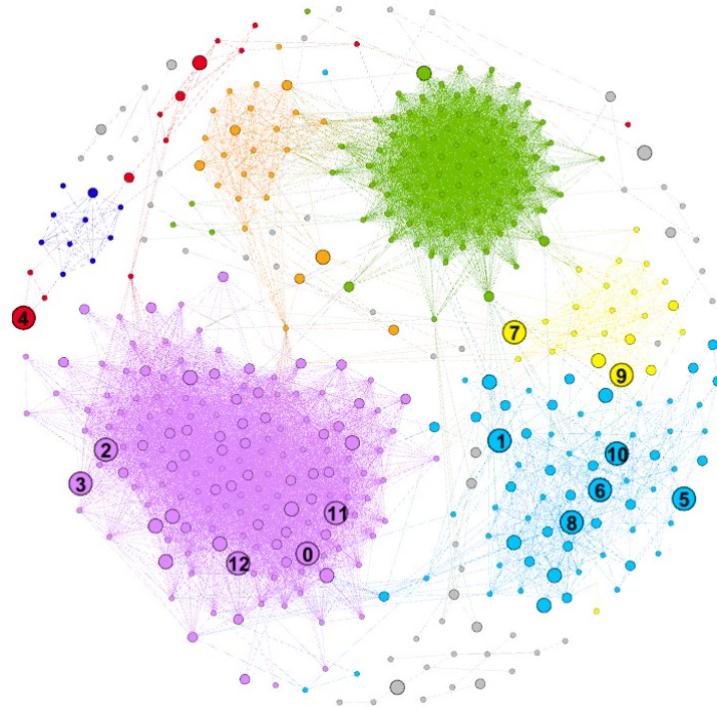


Figure B.8: Bone GCN

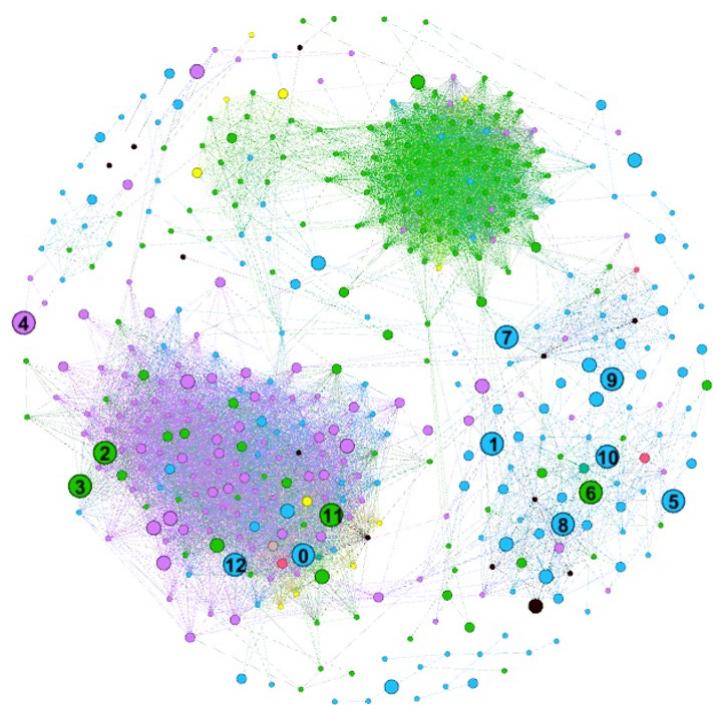


Figure B.9: Bone GCN coloured by Liver GCN's community colour scheme

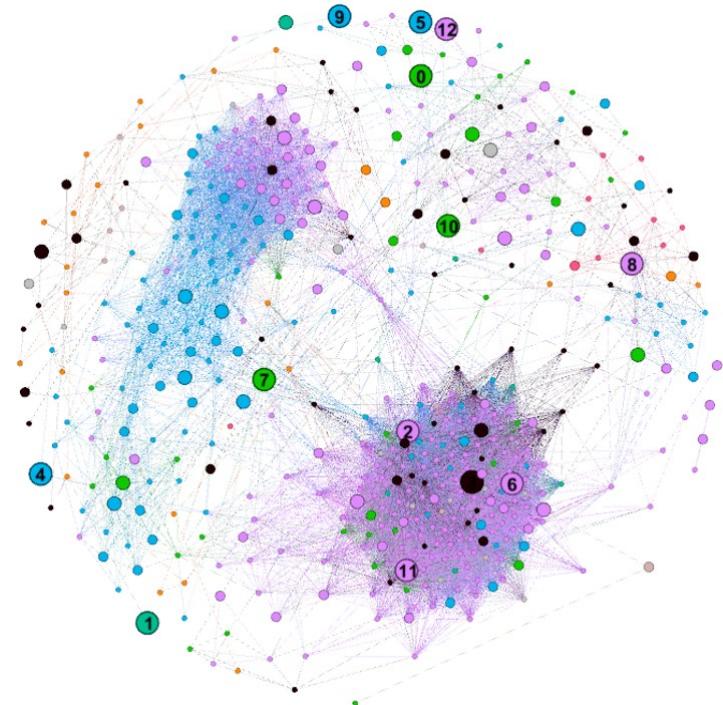


Figure B.10: Liver GCN coloured by LN GCN's community colour scheme

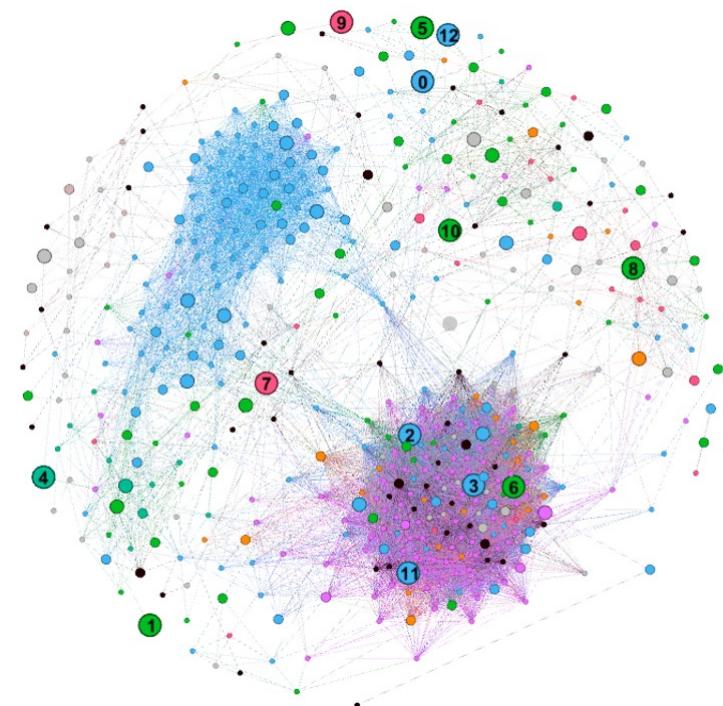


Figure B.11: Liver GCN coloured by Bone GCN's community colour scheme

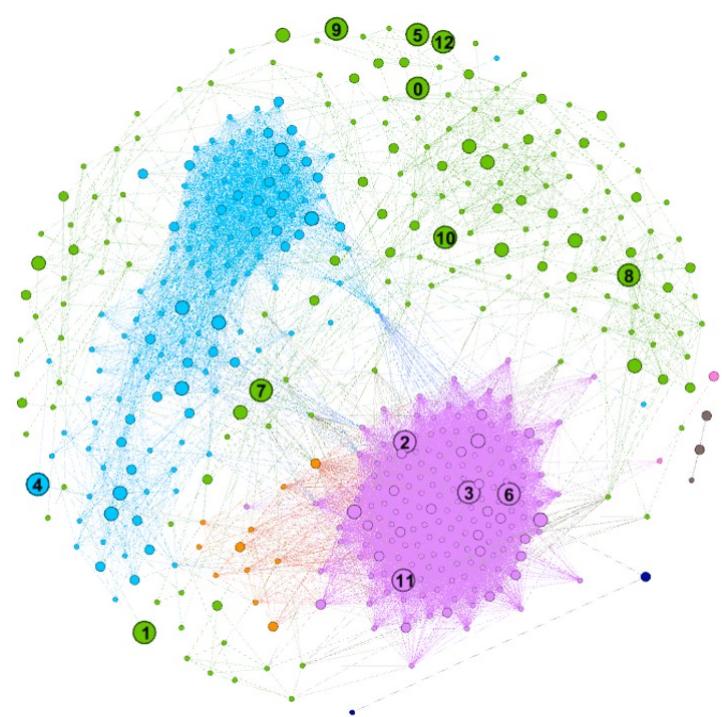


Figure B.12: Liver GCN

Appendix C

Code and Dataset

The code and datasets related to this study are all available at:
https://github.com/zcablii/cancer_project.