Joel Corser

Using TDA to find trends in cancer data

Abstract

In this paper we examine gene data for lung, pancreatic and prostate cancer by looking at how their gene expression values differ between healthy and sick groups by considering the genes in the sample space(or in other words if there are m genes in n patients there are m points that sit in R^n) of the patients, which allows us to see how the data differs relative to the patients. We look for trends in the data by considering these points as vertices and constructing simplicial complexes out of them. To construct the complex, a lazy witness complex is used which selects a number of genes to serve as "landmarks", which are ran for an interval of landmark values. After analyzing the persistence of these we get loops, whose members serve as potential trends in the cancer patient. For the pancreatic, prostate and lung cancer there were 30 genes found with 14 linked to cancer, 47 members were found and 30 members were found with 19 being linked to cancer respectively.

Introduction

In this paper we will investigate gene expression values between cancer and healthy samples. The gene data is downloaded from NCBI[9] and is uploaded into matlab as an m x n data matrix, the data is analyzed by considering how gene expression values differ between patients, so if there are m genes and n patients then there will be m points sitting in R^n space. Firstly, we will discuss how the data will be treated as vertices and formed into a complex which can be analyzed using persistent homology. To construct this complex, we will use lazy witness complexes by selecting a subset of genes that will be landmarks, where the first gene is selected randomly from the dataset and the rest are found deterministically with sequential max min. Moreover, we will discuss the variables that are arbitrary in the formation of these complexes and what their optimal values were. Afterwards, using Javaplex the topological persistence of these complexes will be analyzed and the longest loop will be chosen, where its loop members correspond to potential trends between cancer and non cancer patients. Secondly, the loop members will be confirmed as trends in the data by comparing data between healthy and sick samples using basic statistics, which will outline differences in gene expression values. Lastly, the genes that have been found will be looked up in biomedical literature to confirm whether they contribute to cancer biogenesis.

Assumptions

In this section we briefly discuss assumptions made on the process and data. First we assume the data sampling is consistent between the patients, or that the method to measure gene expression values doesn't introduce a new variable that creates a discrepancy between multiple patients, which could contribute more to noise or false trends(loops). Secondly, we assume that if a trend in the data is significant it will persist across many different landmark values or different choices of a first landmark. Thirdly, we assume that loops found in the data will not be formed after an arbitrary filtration radius, that is they won't form when the data is nearly becoming fully connected, this is used in the case that the distances between the landmarks exceed a computationally reasonable amount. Fourthly, we assume there are enough samples provided.

Our assumptions limit us to finding loops that will persist across random 1st choices of the landmark that occur in intervals less than the arbitrary amount inputted and that these loops will occur in an interval of landmark values that are arbitrary inputted. A way to increase generality and avoid some of these assumptions would be to use parallel processing on a supercomputing cluster and run simulations for all genes as first landmarks across a wide range of landmark values. However, it's unlikely to find more loop members using this process and wouldn't be worth the resources. The arbitrary radius requirement cannot be removed, as it increases computational complexity exponentially which can make it infeasible. Moreover, our assumptions imply that we cannot guarantee all trends will be found, but rather ones that are subject to the prior conditions

Constructing the complexes from gene expression data

This section will discuss how the complexes will be constructed from raw gene expression data. First we upload the data from a soft file into matlab with the bioinformatics tool box, it is uploaded as a field array with multiple elements, and the gene data is extracted from this. The gene data is represented as an m x n matrix, where there are m genes and n patients. The data is analyzed by considering the genes in the patient sample space, this analyzes trends between the patients, rather than trends between the genes themselves, which is the goal since we're trying to find genes that could be carcinogenic. This gives m data points that sit in R^n, which reduces the dimension of the data. For instance, if there are 50000 genes between 100 patients, they now sit in R^{100} as 50000 data points, rather than 100 data points in R^{50000}.

Since we have many data points we will use lazy witness complexes, since VR and alpha complexes will be impractical or computationally intensive. The method for constructing the complexes and analyzing persistence is based on [3]'s approach. The approach creates a complex by picking landmarks(genes) from the data set, that will create a complex based on a given radius or distance. The first landmark has to be chosen randomly while the rest are determined with sequential max min. Sequential max min is deterministic, in the sense that it picks points that maximize the distance from each other, providing more "coverage" when compared to random point sampling.

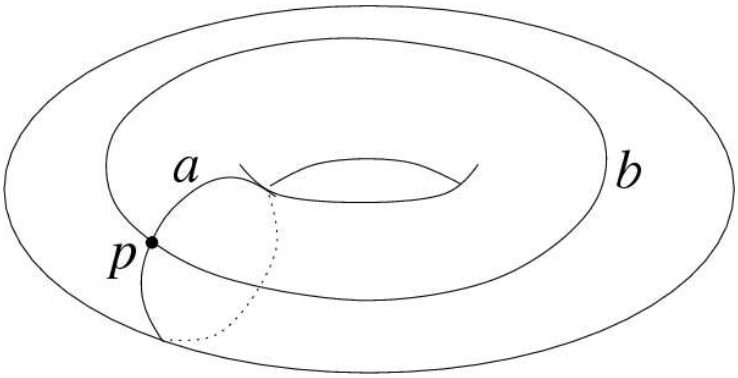
There are a few arbitrary parameters in this process that must be decided beforehand, we need to know which landmark values to use, the maximum filtration radius and the smallest allowable loop.(Loops with intervals shorter than this are dismissed as noise). It is important to understand that too few landmark values will not allow significant features to become connected and they will likely consist of fewer noisy loops, while higher number of landmark values could cause the data to become too connected and "destroy" or fill in any trends that we're investigating. Considering that every data set has the potential to behave differently, we will have to find a good starting point for landmark values, one that creates some loops longer than the noisy ones and one value that is much larger that also creates only smaller noisy loops. The idea is that the significant features in the data we're looking for will form as loops between these landmark values. In other words, we run many simulations across multiple landmark values in an interval [i1,i2] where i1 is the minimum amount to create loops that are larger than the noisy ones, while i2 is the highest value that can create large enough loops. The idea is that the significant features we're looking for will fall within this interval and will reach a maximal size at a "optimal" landmark value and will decrease in size after it. This interval helps to capture potential loops that may not occur at one particular landmark value.

Another issue is that the first landmark is chosen randomly. To help avoid the effects of randomly selecting the first landmark, we will rerun the algorithm around 10-20 times and collect all the results together. For example, if we have to run the algorithm for i=80 to i=200 with di=20 and 20 reruns we effectively run the algorithm 120 times which takes around an hour. It is important to recognize that it is possible for loops to fall outside of our intervals or even form for

another random 1st landmark, so we cannot guarantee we can find all trends. It is recommended to run the algorithm for various values, and determine the landmark interval and rerun count based on the behavior of the data. Some examples of how data can behave differently is shown in the next section.

finding loops in the complexes and what they represent

In this section we discuss what loop members in the complex represent and which loops are significant and how the behavior of the data determines optimal values for the landmark count, reruns and loop interval length. Loops are roughly speaking topological constructs that are connected in some sort of way, whose members consist of vertices of the set. When forming complexes out of the cancer data, the loops members are genes that are connected or follow some sort of trend between the patients. Many loops will form due to the large number of vertices, however many of them will be noise that we will dismiss. The noisy loops correspond to having weaker topological features by having shorter intervals or less persistence, so we ignore these loops and focus on the longer loops after finding a parameter for how long we want the loops. For example, we can consider the vertices of a torus while adding some extra vertices caused by "noise" and then triangulate this into a complex. After analyzing the persistence of this complex, one could find a few smaller loops whose members consist of mostly noisy points and the infinitely persistent loops a and b shown in the figure. We will ignore the shorter loops based on a tolerance and focus our attention on the longer loops a,b which allows us to analyze noisy data without removing essential topological features. As mentioned earlier, a parameter we must know beforehand is what length of intervals are considered strong topological features, this is dependent on the data and must be determined after running the algorithm many times and comparing the longer loops to the shorter ones.



Once we have the parameters for running the algorithm(landmark values, reruns and radius) we will save all the loops formed during these reruns and will look at the members of the longest loops formed and will pool all their data together. In most data sets, the loops will typically be smaller at a lower landmark value and will evolve across increasing landmark values until they reach their maximal size and will decrease afterwards, this is demonstrated in the matrix below for the pancreatic cancer dataset. The point at which they are consistently formed with the longest interval length will be called the "optimal" number of landmarks. However, it is possible for bigger loops to form outside of this, as shown below in the matrix where the j'th entry denotes the landmark value and the i'th entry represents rerun number. The entries represent

interval lengths for the longest loop and the one circled in orange is the longest loop, and the columns circled in red are the optimal landmark numbers, and loops circled in black are ones that we're interested in. Notice how the longest loops gradually increase in size and peak around 9,10 on average and gradually decrease as the number increases.

Prostate cancer loop lengths [80,240] landmark points with di=20

80	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	240 16
1	167.6110	645.8338	782.6401	751.2135	725.8058	697.9652	1.0954e+03	392.0460	767.3129	1.2493e+03	601.9697	719.4857	703.6725	801.1623	675.3293	775.2532
2	330.8174	645.8338	783.0309	751.2135	1.0080e+03	678.5910	405.6545	642.3407	1.4115e+03	1.2483e+03	734.0407	719.4857	703.6725	691.2412	675.3293	775.2532
3	167.6110	802.5317	765.0923	1.0283e+03	725.8058	1.1059e+03	409.7490	392.0460	1.4115e+03	493.2736	734.0407	719.4857	703.6725	691.2412	559.1747	775.2532
4	167.6110	160.7801	784.7918	751.2135	725.8058	697.9652	405.6545	392.0460	383.5343	1.2493e+03	734.0407	717.6695	[illegible]	455.5060	679.0229	775.2532
5	167.6110	642.0254	784.7918	1.0315e+03	727.1691	697.9652	405.6545	392.0460	1.4115e+03	490.9896	734.0407	717.669	2.3244e+03	691.2412	668.7388	775.2532
6	167.6110	645.8338	753.7353	751.2135	725.8058	697.9652	405.6545	392.0460	1.4115e+03	1.2493e+03	734.0407	719.485	[illegible]	691.2412	679.2772	775.2532
7	330.8174	645.8338	782.4456	1.1691e+03	864.0265	697.9652	405.6545	392.0460	1.4067e+03	1.2483e+03	734.0407	719.4857	703.6725	691.2412	676.2963	774.4022
8	330.8174	642.0254	782.4456	751.2135	725.8058	697.9652	405.6545	514.3522	1.4115e+03	491.3727	734.0407	719.4857	703.6725	456.4057	679.2772	774.4022
9	167.6110	645.8338	784.7918	751.2135	727.1691	697.9652	271.1084	258.5383	1.4067e+03	1.2493e+03	484.1250	591.6208	703.6725	452.8515	679.0229	775.2532
10	167.6110	645.8338	783.8531	1.0315e+03	725.8058	697.9652	405.6545	392.0460	1.4115e+03	1.2493e+03	734.0407	475.6276	691.9945	691.2412	679.0229	553.7523
11	167.6110	644.2350	784.7918	744.583	1.0106e+03	697.9652	405.6545	394.2463	381.6802	1.2493e+03	484.4232	717.6898	703.6682	454.9599	556.6122	775.2532
12	331.9721	645.8338	908.7026	751.2135	725.8058	697.9652	406.6626	392.0460	1.4067e+03	742.6419	484.4232	475.3856	467.7171	691.2412	1.5755e+03	775.2532
13	328.3816	645.8338	782.4456	751.2135	727.1691	697.1082	532.0514	918.6980	381.6802	490.9896	734.0407	719.4857	703.6725	691.2412	677.6284	775.2532
14	674.2396	642.0254	784.7918	751.2135	725.8058	697.9652	405.6545	392.0460	1.4115e+03	493.2736	734.0407	719.4857	465.7395	691.2412	679.0229	774.4022
15	167.6110	645.8338	784.7918	751.2135	1.0080e+03	697.1082	405.6545	392.0460	1.4115e+03	1.2493e+03	734.0407	711.7988	703.6725	691.2412	679.0229	774.4022
16	498.6997	645.8338	782.4456	751.2135	725.8058	697.9652	405.6545	392.0460	1.4115e+03	1.2493e+03	734.0407	717.6695	703.6725	691.2412	679.0229	775.2532
17	167.6110	320.9518	784.7918	744.5835	725.8058	972.6962	405.6545	392.0460	381.6802	1.2493e+03	734.0407	468.7964	703.6725	456.4057	677.6284	552.9449
18	167.6110	642.0254	782.445	1.0339e+03	725.8058	697.9652	809.1290	392.0460	383.6565	1.2493e+03	484.4232	717.6695	703.6725	681.3625	679.1370	661.4937

In this dataset we also set our interval length parameter too low, resulting in a few extra loop members that were likely noise. In this case it would have been better to set it to exclude the loops in columns 4-7 and focus on 9,10 and the loop in 13 and 15. Even though some entries have the same value, we still check all of them in case another slightly smaller loop forms, which was the case for the pancreatic cancer and lung cancer data. Looking at the pancreatic cancer data below, one can observe the optimal landmark number is 2 and that picking the 1st landmark doesn't affect the results much, unlike the prostate cancer results above, which show variance when picking different starting landmarks.

Pancreatic cancer loop lengths [100,180] landmark points with di=20

	1	2	3	4
1	12400	22800	13200	14800
2	10800	22800	12000	10400
3	10800	22800	13200	10400
4	10800	22800	Inf	11600
5	10800	22800	13200	10400

We investigated only members from the second column and found many loops, most of the loops were secondary slightly smaller loops that would form and disappear on reruns. The pancreatic cancer data appeared to be "well behaved" not being sensitive to the first landmark and evolving with landmark values in the expected manner. Sometimes data sets don't behave

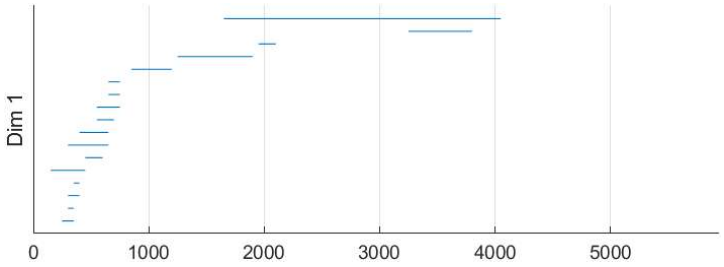
like the ones shown above and don't form a majority of their longest loops at the optimal landmark number as shown below. Some are more sensitive to the choice of the first landmark and various landmark values as shown below. The lung cancer dataset wasn't very consistent with how the loops change with landmark values and was showed sensitivity to the selection of the first landmark.In this case it would have been better to run for a wide interval of values and many reruns to make sure we don't miss as many longer loops and keep the di parameter low(around 5-15). The interval for landmark values was [160,220] with di=20. Values to the left were much smaller and didn't contain any loops we were interested in. It is possible we could have missed many important loops by limiting it to [160,220].

Lung cancer interval lengths [160,220] landmark points with DI=20

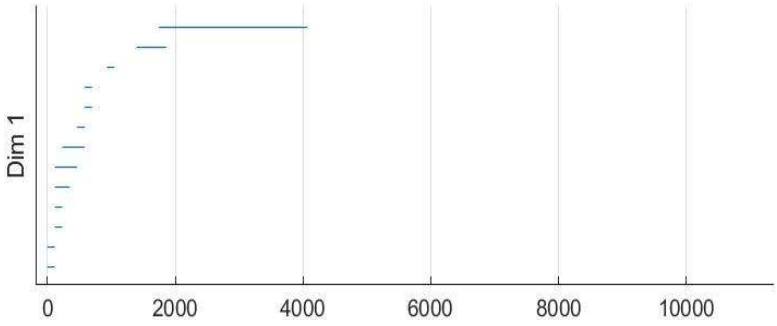
	1	2	3	4	5
1	1591	774	1419	1161	
2	1935	1333	1419	1892	
3	1591	1075	Inf	1161	
4	1591	774	1419	1161	
5	817	774	1419	1892	
6	1591	774	2193	1892	
7	1591	774	1892	1161	
8	1591	1075	1419	1161	
9	1591	817	1419	1161	
10	1591	817	1419	1161	
11	2322	1376	1462	1161	
12	1591	2193	1419	1161	
13	1591	817	1419	1161	
14	1591	1075	1419	1892	
15	817	1075	1419	1161	
16	1591	817	1462	1161	
17	1591	2193	1419	1161	
18	1591	2193	1419	1161	
19	1591	817	1763	1634	
20	1591	817	1419	1161	
21					

The longest loops found in the previous matrices are shown below in their barcode diagram

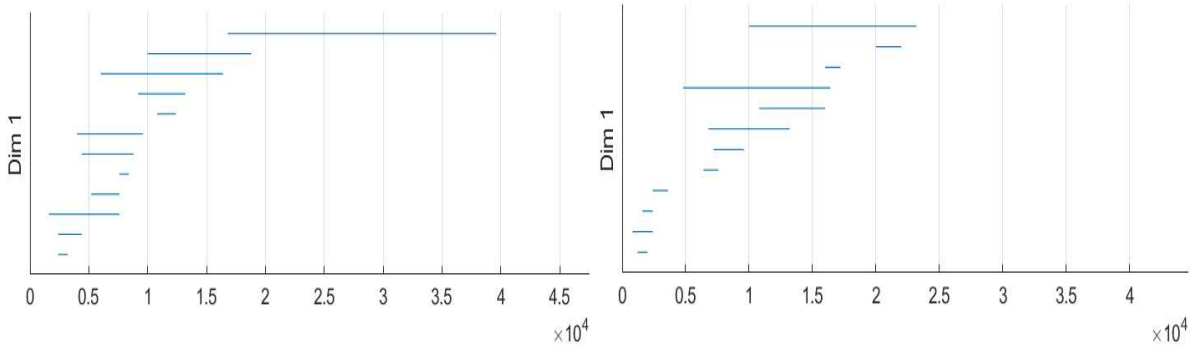
Lung



Prostate



Pancreas loop pair(there were many others this is just an example of two)



Analyzing loop members

This section will analyze which genes are loop members and whether or not they are statistically significant and if they are a likely cause for cancer. In order to analyze whether the genes correlate to the cancer or not we consider whether it is statistically significant and if there is a correlation between the genes function and potential for cancer biogenesis. When considering statistical significance we consider the mean and median for the healthy and sick data sets and determine whether there is a difference between the two, or if being sick plays a factor in the gene expression value. When finding the correlation of the genes and cancer, we use genecards or the NIH gene database and list their function to the side.

For the pancreatic cancer dataset we found 30 loop members, 14 have potential for cancer biogenesis and 5 were statistically insignificant or had limited information available and two were controls. Some genes of interest with high differences in expression values from the healthy group were FXYD3, PRNP and EEF1D.

For the lung cancer dataset we found 28 loop members, 19 of these members have potential for cancer biogenesis and 5 were insignificant or had no information available. The most profound difference was in the TXNIP and gene which is involved in tumor suppression.

For the prostate cancer dataset we found 47 loop members, however many of these members were insignificant. It is likely the large amount of insignificant members are a result of having our interval length parameter set too low. A gene with the largest difference in expression values was the MUC4 gene, which has been shown to inhibit cellular apoptosis and promote tumor growth.[1,2] Not all members were researched for their potential for cancer biogenesis as there were too many.

Results for the statistical difference between the healthy and sick groups as well as gene functions are included as charts in the appendix.

Conclusion

In summary we used TDA on 3 different data sets with gene expression values for lung, pancreatic and prostate cancer and found loops whose members correspond to genes of interest. To find the loops, we constructed witness complexes with the first landmark chosen randomly and the remaining calculated with sequential max min. We then analyzed the persistence of these complexes for an interval of landmark values. Loops that were sufficiently large were selected and their members were investigated for their potential for cancer biogenesis. For the lung cancer dataset we found 28 members, while 19 of those members were linked to cancer. For the pancreatic cancer set, we found 28 loop members 14 of were linked to cancer. For the prostate cancer set, we didn't investigate all of the genes correlation to cancer, but we calculated the differences in expression values which are provided in the appendix.

References:

[1] NIH gene database https://www.ncbi.nlm.nih.gov/gene/
[2] https://www.genecards.org/
[3] Lockwood, S., & Krishnamoorthy, B. (2014). Topological Features In Cancer Gene Expression Data. Biocomputing 2015. doi:10.1142/9789814644730_0012
[4] http://appliedtopology.github.io/javaplex/
[5] https://www.math.colostate.edu/~adams/research/javaplex_tutorial.pdf
[6] Meacci, Elisa, et al. “Lung Metastasectomy Following Kidney Tumors: Outcomes and Prognostic Factors from a Single-Center Experience.” *Journal of Thoracic Disease*, vol. 9, no. S12, 2017, doi:10.21037/jtd.2017.05.04.
[7]Pan, Y., Ni, R., Deng, Q., Huang, X., Zhang, Y., Lu, C., . . . Chen, B. (2013). Glyoxylate Reductase/Hydroxypyruvate Reductase: A Novel Prognostic Marker for Hepatocellular Carcinoma Patients after Curative Resection. *Pathobiology, 80*(3), 155-162. doi:10.1159/000346476
[8] Riabov, V., Yin, S., Song, B., Avdic, A., Schledzewski, K., Ovsiy, I., . . . Kzhyshkowska, J. (2016). Stabilin-1 is expressed in human breast cancer and supports tumor growth in mammary adenocarcinoma mouse model. *Oncotarget, 7*(21), 31097-31110. doi:10.18632/oncotarget.8857
[9]https://www.ncbi.nlm.nih.gov/sites/GDSbrowser
[10]https://www.takarabio.com/learning-centers/stem-cell-research/technical-notes/neural-stem-cells/rhb-a-neural-stem-cell-medium#:~:text=The%20RHB%2DA%20system%20enables,and%20differentiation%20of%20NS%20cells.
[11]https://commons.wikimedia.org/wiki/File:Fundamental_group_torus2.png
Where the image of the torus was found.

Appendix

Instructions for code

There are multiple functions used in the code. The most important one obviously is the javaplex package that is installed for matlab. The codes I made were outlined below with their instructions. Codes also have annotations provided. Run the codes in the following order to make sure they work. **1.)** Load_javaplex.m. **2.)** persist.m and the other two will work as they use functions in the former codes.

persist.m

This is the working part of the code that will create a lazy witness complex and will analyze its persistence. It will output the number of simplices, loops and their intervals, interval lengths stored in array form and another array that outlines the index of the loop with the longest interval i. It will take Data, landmark number and maximum radius as inputs in the formpersist(Data,landmarks,maxradius). It is important to be very conservative with radius values, as it will exponentially increase the numbers of simplices performed, which can cause memory errors.

landmarkfinder.m

This scans over an interval of landmark values and creates a matrix showing interval lengths for different landmark values. It is recommended to run this multiple times to check for consistency and select the number of landmarks corresponding to the longest loop and rerun it around 10 times. It will also save all loops in a cell called loopmore. The idea is to get an idea of interval lengths to decide a cutoff for their values, and select the loops you want to investigate. If you are interested in say the value in the matrix that is mxn at i,j with length "2000" to retrieve the loop for this you do loopmore((i-1)*n+j)). There is also a version of this that is parallel, as the whole process is embarrassingly parallel, however due to the way matlab works it will create errors with javaplex as it wont create subprocesses or more threads. A recommended amount of memory for this is around 4gb, to make sure memory errors don't crash the whole loop.

statstuff.m

Simple code, just breaks the data apart and creates histograms and calculates the mean and median. Need the indices of healthy and sick patients, or if it's organized in standard format the number of sick patients and healthy ones, where the first part are healthy samples and the rest are sick or vice versa.

load_javaplex.m

Loads javaplex and runs a simple code. You need to run this one first, the simple code uses different commands from javaplex to make sure it is working without errors, if it works you will see two circles with dots on them.

Lung cancer gene results

Range for landmarks was between 80 and 240

E1F3CL	limited information available
RHB	Enables stable proliferation of some nerve cells[10]
BCAP31	May be involved in apoptosis[2]
PHC2	Involved with repressing some genes[2]
TAX1BP1	Inhibits cellular apoptosis[2]
PLP2	May play a role in cell differentiation[2]
PSMA2	Maintains protein homeostasis and removes/repairs damaged proteins that can impair cellular function[2]
RPA2	Involved in DNA metabolism and replication[2]
SUPT7L	Chromatin organization, involvement with DNA[2]
RYBP	Possible tumor growth repressor[2]
SLCO2B1	increased activity is associated with ilium and breast cancer[2]
VCP	Regulates various proteins, including those involved in cellular division and replication[2]
ARPC2	Promotes cellular motility and repairs damaged DNA[2]
AF130103	No information available[2]
IGFBP5	May inhibit or promote cellular growth depending[2]
TMEM123	Promotes cell death by oncosis[2]
ADO	Linked to throat cancer[2]
NCKAP1	Involved in endocytosis[2]
YWHAB	Inhibits neuronal apoptosis[2]
UBE2Q1	More active in females[2]

RTN3	N/A(statistically insignificant)
WASL	Possible role in gene regulation[2]
SYAP1	Possible role in cell differentiation[2]
LOC1019301	No information available
KHSRP	Involved in degrading RNA[2]
PRKAG2	Regulates cellular metabolism[2]
MALAT1	Regulates various genes that can promote cell growth and proliferation[2]
PTGR1	N/A(<5% difference on mean and median)
TACC3	Might have function with control of cell growth and differentiation. Also likely linked to cancer.[1,2]
TXNIP	Involved in tumor suppression[1,2]

1	53.6745	52.3383	EIF3CL
2	-2.0699	-20.7565	PHB
3	26.3183	28.3840	BCAP31
4	-6.2470	-21.5332	PHC2
5	-24.4511	-12.0970	TAX1BP1
6	29.3797	40.0613	PLP2
7	-15.8788	-13.0093	PSMA2
8	-5.2464	3.2401	RPA2
9	-19.6611	-18.5455	SUPT7L
10	33.3811	36.7666	RYBP
11	57.5016	65.9259	SLCO2B1
12	28.3329	32.0257	VCP
13	51.8521	56.1337	ARPC2
14	35.0930	72.1805	AF130103
15	39.5663	50.7909	IGFBP5
16	-10.0263	-8.4893	TMEM123
17	9.5510	-14.2077	ADO
18	21.4932	48.7622	NCKAP1
19	14.8659	17.6501	YWHAB
20	3.8036	18.5277	UBE2Q1
21	0.8284	5.6910	RTN3
22	-9.9960	-11.2671	WASL
23	-22.3513	-7.7768	SYAP1
24	-11.8561	-12.6270	LOC101930...
25	9.2185	3.9801	KHSRP
26	26.4993	22.2222	PRKAG2
27	-26.2509	-41.3174	MALAT1
28	-10.7978	-1.9987	PTGR1

19 of these loop members have potential for cancer biogenesis, 5 were insignificant or had no information available and 6 likely don't contribute to cancer, they may be a result of having the cancer. The most profound difference was in the TXNIP gene, with a difference between the mean and median being -152%/-157%. The insignificant loops were likely from including extra loops just in case they had potential members.

The calculated percent difference from the healthy group is shown in the table to the left, where mean is the 1st column and median is the second column.

Pancreatic cancer results(Used landmarks 100-180 while using 120 as the center)

Gene	Function
CBX3	N/A
SNX3	Plays a role in protein transport in cells[2]
ZYX	Mediates the formation of adhesions. Correlated with some tumors such as sarcoma.[2]
PRNP	Unclear function, but related to prion diseases(non cancerous). May induce apoptosis in some cells[2]
RPA2	Involved in DNA metabolism and replication, was also seen in the lung cancer results as a loop member.[2]
DVL3	Regulates cell proliferation[2]
AFF1	seen in leukemia and a few other cancers. Function has something to do with transcriptional misregulation of cancer.[2]
FXYD3	May contribute to tumor progression[2]
EEF1D	Used to repress protein production in host cells[2]
KCNJ8	Has some function in potassium regulation in cells[2]
SLC22A2	Helps assist in an anti cancer and tumor process.[2]
FGF8	Supports cell growth, proliferation and development. Normally expressed in reproductive organs in mammals.[2]
RPS14	Anomalies in this gene are associated with resistance to a protein synthesis inhibitor[2]
LUC7L3	Associated with diabetes. Possibly over expressed from a tumor damaging the pancreas.[2]
SNORD83B	Used in RNA processing and modification [2]
SMARCE1	N/A(less than 5% difference on
EXOC7	Has a function with insulin interacting with cells[2]
CLCC1	Not much info available, seems to have something to do with a sodium channel[2]
GJC2	Used in diffusion of molecules[2]
SNTB2	Used in the process of creating dystrophin [2]

X86400	No information available
KPTN	May be involved in the downregulation of a gene (mTORC1) whose downregulation has been linked to cancer[2]
MALAT1	Regulates various genes that can promote cell growth and proliferation. Was also seen in the lung cancer results[2]
TMED8	Tracking protein, limited info[2]
TMEM87B	not much info on its function, but is related to 2q13 deletion syndrome[2]
DUSP18	Used in dephosphorylation[2]
ZNF83	May be involved in transcriptional regulation[2]
AI057052	No information available
ATOX1	Plays a significant role in cancer carcinogenesis[2](this is an extra one found that is not included in the table below)

	1	2	Gene
1	11.3087	4.2762	CBX3
2	-29.5954	-39.3974	SNX3
3	-39.2415	-57.4646	ZYX
4	-86.1755	-119.7853	PRNP
5	-18.2932	-23.6897	RPA2
6	-35.4148	-30.8310	DVL3
7	29.4951	19.9719	AFF1
8	130.3651	123.4725	FXYD3
9	-119.9565	-132.4537	EEF1D
10	-176.6658	-193.9154	KCNJ8
11	-22.4959	-52.2906	SLC22A2
12	4.7411	-71.4859	FGF8
13	-94.5352	-80.8661	RPS14
14	53.9007	52.9312	LUC7L3
15	-71.0666	-72.1491	SNORD83B
16	3.2412	2.0076	SMARCE1
17	-46.6863	-39.3758	EXOC7
18	-23.0181	-16.7520	CLCC1
19	-6.8932	-29.8901	GJC2
20	-43.3152	7.0866	SNTB2
21	-37.9293	-54.2983	X86400
22	16.6866	35.1810	KPTN
23	34.8578	23.3431	MALAT1
24	-11.5838	-17.1051	TMED8
25	85.0432	95.0342	TMEM87B
26	15.9962	11.6673	DUSP18
27	66.1848	25.8720	ZNF83
28	23.7719	34.8870	AI057052
29	-30.6056	-14.1360	--Control
30	-55.4738	-29.8582	--Control

14 of the genes found contribute to cancer biogenesis, while 5 were statistically insignificant or had no information available, two were controls and the rest were likely results of having the cancer. FXYD3, PRNP and EEF1D were the cancer related genes with the largest difference between the sick and healthy groups.

The calculated percent difference from the healthy group is shown in the table to the left, where mean is the 1st column and median is the second column.

	1	2		1
1	4.2056	13.5451	1	ARL17B
2	2.3901	11.0039	2	LOC101927...
3	8.3464	-0.3354	3	CTDNEP1
4	22.3295	15.8793	4	SSR2
5	0.6751	5.4779	5	SUMO3
6	-8.6124	46.9516	6	CALR
7	9.2485	8.0179	7	SSR4
8	4.5253	-13.1845	8	TXNIP
9	-31.9825	-32.0782	9	DSTN
10	-20.0631	-18.8949	10	RAB5C
11	-10.0113	-8.4876	11	G3BP1
12	-0.4312	-2.8411	12	UBE2N
13	1.3523	-2.0255	13	ZC3H15
14	-20.5769	-7.0088	14	DHCR7
15	10.2659	-0.8245	15	ACP2
16	13.9180	19.8232	16	RGS3
17	-7.3823	-9.1048	17	SKAP2
18	28.5423	35.8856	18	KIF21B
19	-5.1731	-5.5564	19	PFN2
20	8.7196	11.2177	20	KISS1
21	-24.0988	-48.6565	21	F8
22	-37.8384	-65.6741	22	ZNF702P
23	4.8054	6.5574	23	RNF10
24	-12.3441	2.7096	24	PPP6R1
25	-7.9027	-3.4360	25	BBS9
26	-36.3676	-17.9332	26	ACTN1
27	-39.7324	-43.0656	27	PDE4B
28	34.5556	27.8168	28	GGTLC1
29	0.2278	-1.2478	29	GH2
30	-14.0708	3.6035	30	U87229
31	17.1921	8.9676	31	CFB
32	-39.3374	-94.2394	32	MUC4
33	11.6050	5.2140	33	TSSK2
34	4.6275	17.8680	34	MIR1282
35	16.2409	8.3852	35	NT5C3A
36	7.5556	20.1650	36	MIR612
37	-2.8336	0.5596	37	TMEM230
38	-31.1423	-29.3613	38	C4orf3
39	-2.7419	-1.2819	39	GFM1
40	-0.0146	-10.5528	40	MSI2
41	-26.5347	-42.1835	41	ATP8B1
42	28.9053	7.8841	42	DTX1
43	2.4031	17.3407	43	CIRBP
44	-15.9494	-17.0465	44	C8orf46
45	17.1452	25.1442	45	TBCD
46	-8.7896	-12.1518	46	FAM168A
47	16.1472	17.1200	47	--Control

Prostate cancer genes

The calculated percent difference from the healthy group is shown in the table to the left, where mean is the 1st column and median is the second column.