Title of My Thesis

A thesis

presented to

the faculty of the Department of Mathematics

East Tennessee State University

In partial fulfillment

of the requirements for the degree

Master of Science in Mathematical Sciences

---

by

Bob Gardner

May 2007

Bob Price, Ph.D., Chair

Anant Godbole, Ph.D.

Debra Knisley, Ph.D.

Keywords: graph theory, algebra, complex analysis, statistics

PLAIN LANGUAGE ABSTRACT

All manuscripts are now required to have a plain language abstract in addition to the regular abstract. A plain language abstract for a thesis is strictly limited to 150 words or less, 350 words or less for a dissertation. A plain language abstract is a short, simple explanation of your research. It should not use jargon or technical terms. It should be understandable by an average member of the public. It should simply explain why the research was done, how it was done, and the key findings. The purpose is to make your research more accessible to more people. This is especially important for reaching a broader audience, including policymakers, journalists, and the public.

For additional help, go to the first link under the Writing Help tab on the Resources page at https://www.etsu.edu/gradschool/etd/resources.php

ABSTRACT

Title of My Thesis

by

Bob Gardner

Insert text of abstract here.

Copyright by Bob Gardner 2007

All Rights Reserved

GENERATIVE ARTIFICIAL INTELLIGENCE DISCLOSURE

Generative Artificial Intelligence (GAI) tools, including [specific tools], were utilized in this [capstone project/thesis/dissertation/synthesis project] for [specific uses]. While AI-generated outputs were reviewed for accuracy, relevance, and integrity to align with academic standards, I am responsible for the integrity, originality, and academic validity of this scholarly work.

DEDICATION

Text of Dedication.

ACKNOWLEDGMENTS

Insert text of acknowledgements here.

TABLE OF CONTENTS

PLAIN LANGUAGE ABSTRACT 2

ABSTRACT 3

GENERATIVE ARTIFICIAL INTELLIGENCE DISCLOSURE 5

DEDICATION 6

ACKNOWLEDGMENTS 7

LIST OF TABLES 9

LIST OF FIGURES 10

1 TITLE OF FIRST CHAPTER 11

1.1 Subsection Title 11

2 TITLE OF SECOND CHAPTER 13

BIBLIOGRAPHY 14

APPENDICES 15

A Code Implementation of 15

A.1 Embedded Code Example – Python 15

A.2 Imported Code Example – Python 15

B Appendix 2 Title. 17

B.1 First Subsection of Appendix 2 17

VITA 18

LIST OF TABLES

1 Caption for the first table. 11

LIST OF FIGURES

1 Caption for the first figure. 13

2 Caption for the second figure. 13

1 TITLE OF FIRST CHAPTER

Test of Chapter 1.

1.1 Subsection Title

Text of subsection.

Theorem 1.1 [2] *A bounded function f is Riemann integrable on $[a, b]$ if and only if the set of discontinuities of f has measure* 0.

An example of citing an article: It is a small world, after all [4].

A great paper to read for autoencoders is [3]. A great paper to read for Koopman theory is [1, 3].

Table 1: Caption for the first table.

Column 1	Column 2
1	2
3	4

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud

exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Text of Chapter 2, which continues [1].

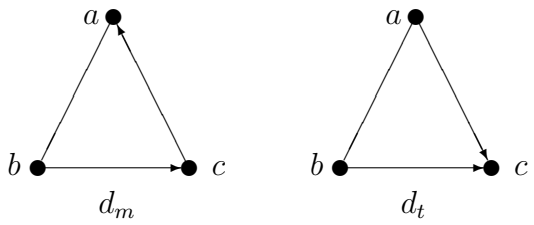

Figure 1: Caption for the first figure.

Figure 2: Caption for the second figure.

BIBLIOGRAPHY

[1] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, 113(15):3932–3937, 2016.

[2] J. R. Kirkwood. *An Introduction to Analysis*. PWS Publishing Company and Waveland Press, 2 edition, 1995.

[3] Andrew Ng et al. Sparse autoencoder. *CS294A Lecture notes*, 72(2011):1–19, 2011.

[4] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world' networks. *nature*, 393(6684):440–442, 1998.

APPENDICES

A Code Implementation of ...

Text of Appendix 1.

A.1 Embedded Code Example – Python

```
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int)  #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

                if M is None:
                    M = np.copy(VT)
                else:
                    M = np.concatenate((M, VT), 1)

                VT = np.zeros((n*m,1), int)

    return M
```

A.2 Imported Code Example – Python

```
import matplotlib.pyplot as plt
import numpy as np
```

```
from matplotlib import Path

fig,axes = plt.subplots()

#function for generating vertices randomly
def RandomVertices(n = 3):
    '''Default is a random triangle in the unit square'''
    x,y = np.random.rand((2,n))
    return [(x[i],y[i]) for i in range(n)]

#plot the Feasible Region
vertices = [ (0,0), (15,0), (15,5), (10,10), (6,12)  ]
#Uncomment next for random vertices
#vertices = RandomVertices(n = 5)

X,Y = zip(*vertices)
axes.fill(X,Y, facecolor='cyan', edgecolor='black', zorder
    = 1)
Feasible = Path( vertices )


#plot the vertices
axes.scatter(X,Y, color='brown', zorder = 2)
for vert in vertices:
    axes.text( vert[0], vert[1], " ({0},{1})".format(vert
       [0], vert[1]), fontsize = 14)

# Label the resulting feasible region
axes.text(7,3, '$\Omega$', fontsize = 14 )

# Plot the xy-axes and set upper bound on y
axes.set_xlim(0,20)
axes.set_ylim(0,15)
axes.set_xlabel("$p_1$ = 1000's of Standard shafts\n")
axes.set_ylabel("$p_2$ = 1000's of Engineering shafts")
```

B Appendix 2 Title.

B.1 First Subsection of Appendix 2

Text of Appendix 2.

VITA

BOB GARDNER

Education:	B.S. Mathematics, Auburn University in Montgomery, Montgomery, Alabama 1984 M.S. Mathematics (Combinatorics), Auburn University, Auburn, Alabama 1987 M.S. Zoology and Wildlife (Population Genetics), Auburn University, Auburn, Alabama 1992 Ph.D. Mathematics (Complex Analysis), Auburn University, Auburn, Alabama 1991
Professional Experience:	Assistant Professor, LSU in Shreveport, Shreveport, Louisiana, 1991–1993 Assistant, Associate, Full Professor, East Tennessee State University, Johnson City, Tennessee, 1993–present
Publications:	R. Gardner and N. K. Govil, "Some Inequalities for Entire Functions of Exponential Type," *Proceedings of the American Mathematical, Society* **123**(9) (1995) 2757–2761.