

Data driven machine learning-based Steel tension members failure mode probability prediction

***A B. Tech Project Report Submitted
in Fulfillment of the Requirements
for the Degree of***

Bachelor of Technology

by

Gaurav Singh

(101701010)

under the guidance of

Dr. Gokulnath C.



IIT PALAKKAD

**DEPARTMENT OF CIVIL ENGINEERING INDIAN INSTITUTE OF
TECHNOLOGY PALAKKAD**

CERTIFICATE

*I hereby certify that the work contained in this report titled “**Data driven machine learning-based Steel tension members failure mode probability prediction**” Is my bonafide work, carried out in partial fulfillment for the Degree of Bachelor of Technology in Civil Engineering at Indian Institute of Technology Palakkad and that it has not been submitted elsewhere for a degree.*

Gaurav Singh

101701010

Civil Engineering

Indian Institute of Technology Palakkad

May 2021

Acknowledgements

Completing a task is never a one- man effort. Contribution a number of individuals plays a major role in a direct or indirect manner. It brings me great pleasure and immense satisfaction to express my sincere and deepest sense of profound gratitude towards Dr. Gokul Nath C. for giving the opportunity to learn many new things through this project and helping me guide along with it. I am also thankful to all the professors who have been of help for acquiring the practical knowledge of civil engineering works.

Contents

List of Figures	6
1 Introduction	7
1.1 Motivation	7
1.2 Objectives	8
1.3 IS-800 codal provision	9
1.4 Organization of the report.....	10
2 Exploratory data analysis	11
2.1 Feature selection	11
2.2 Pair-plots	17
2.3 Outlier treatment.....	19
2.4 Variable encoding and data-scaling.....	21
3 Building the model	22
3.1 Logistic regression	22
3.2 Support vector machine.....	29
3.3 Decision trees and random forest	32
3.4 K-nearest neighbors and naïve bayes.....	36
4 Conclusion and future work	41
4.1 Conclusion	42
4.2 Future work	37
References	41

List of Figures

1.1 Failure mode in steel tension members	
a. Gross failure, b. Net failure, c. Block shear failure.....	8
1.2 Sections used for the study	
a. Single angle b. Double angle c. T-section d. wide W section.....	10
2.1 The prepared dataset	
a. multi-feature dataset b. Area feature dataset.....	16
2.2 Correlation.....	16
2.3 Correlation values of features.....	17
2.4 Pair-plots for 1 st dataset.....	18
2.5 Pair-plots for 2 nd dataset.....	19
2.6 Outliers.....	19
2.7 Z-score, box plot, and scatter plot of unconnected leg thickness.....	20
2.8 Variable encodings of the sections.....	21
3.1 Sigmoid function.....	22
3.2 Gradient descent.....	23
3.3 Ago Vs Anc.....	24
3.4 Decision boundary of linear logistic regression.....	25
3.5 Confusion matrix.....	25
3.6 Confusion matrix of single feature linear logistic regression.....	26
3.7 Single feature non-linear logistic regression a. Confusion matrix b. quadratic decision boundary.....	27
3.8 multi-feature logistic classifier a. Training data b. testing data.....	28
3.9 Large margin classification decision boundary with largest margin.....	29
3.10 Svm for single feature a. Decision boundary b. confusion matrix.....	31
3.11 multi-feature Svm classifier a. Training data b. testing data.....	31
3.12 Decision tree structure.....	32
3.13 Decision boundary of single feature decision tree and random forest.....	34
3.14 Decision tree representation.....	35
3.15 Random Forest trees.....	36
3.16 K-nearest neighbor classifier.....	37
3.17 Decision boundary for KNN and naïve bayes single feature classifier a. KNN b. naïve bayes.....	39

3.18 Single-feature KNN and naïve bayes classifier a. KNN b. naïve bayes.....	39
3.19 multi-feature KNN and naïve bayes classifier a. KNN-Training data b. KNN-testing data	
a. Naïve bayes training data	
b. Naïve bayes testing data.....	40

Chapter 1 Introduction

1.1 Motivation:

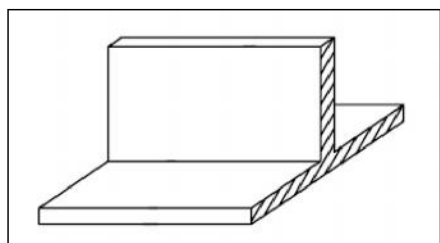
Steel structure is a metal structure which is made of structural steel components connected with each other to carry loads and provide full rigidity. Because of the high strength grade of steel, this structure is reliable and requires less raw materials than other types of structure like concrete structure and timber structure. Steel is now one the most widely used construction material due to these properties. Examples of structural steel include frame structures, beams, columns, grid structures, prestressed, truss members etc. these structural members can be acted upon either tensile forces or compressive, we need to design them accordingly. Tension members are linear members in which axial forces act to cause elongation (stretch). A tension member may have bolted or welded end connections.

The manner of jointing the member to other parts will influence the manner in which the tensile force is transferred into the member. In the case of fastening by bolts, the presence of holes in the section of the member has a direct effect on the strength of the member. We consider this problem by knowing its gross and net sectional area. The gross area refers to the original cross-sectional area of the member and the net area refers to the reduced sectional area after deductions due to the presence of bolt holes. The effective sectional area of a tension member is less than its gross-sectional area due to bolt holes. A tension member undergoes elongation and can extend until it reaches its ultimate strength. As the tensile load reaches the ultimate load the member reaches the failure point. A member in tension can reach a failure state due to excessive elongation or by rupture of its section. The member may become non-functional due to excessive elongation. It is known that this rupture of the section can occur in 2 ways, a Net failure or block shear failure. However, since it is desirable that the failure occurs as a ductile failure rather than as a brittle failure at collapse state. From this point of view, it is desirable to ensure that the gross-section yields before the net section reaches the ultimate stress. So due to these reasons it is important to know the failure mode of tension member for designing the section. Section 6 of IS-800 2007 tells us a method to determine the failure mode, but rather in this study the data is collected from various research papers about section failure mode in tension members and various machine learning models are developed to predict the failure mode probability with given sectional properties.

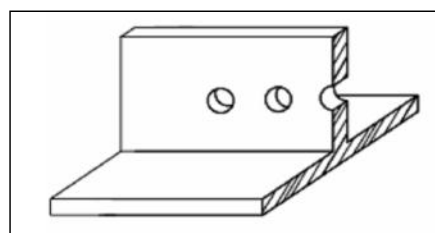
1.2 Objectives:

The main aim of the study is to show that failure mode classification can also be done by using data-driven machine learning based methods, some of the objectives of the study are:

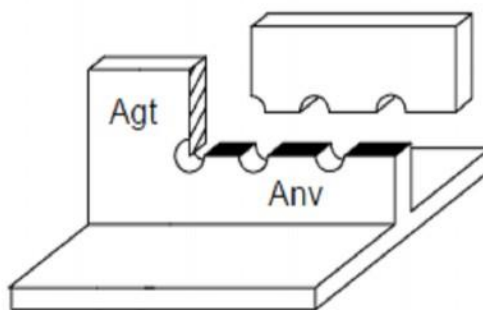
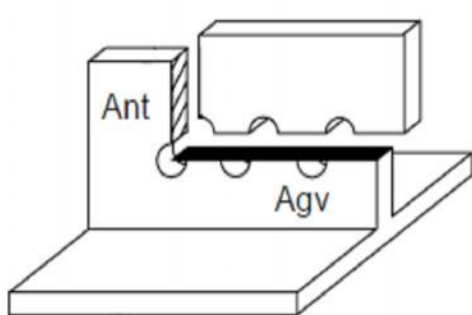
1. Firstly, developing the machine learning models for multiple features (connected leg length, thickness, pitch, gauge, f_y , f_u , bolt details etc.) and comparing the accuracy of developed models.
2. Second, using only the sectional area parameters (A_g , A_n etc. as specified by the IS-800 code) to develop a smart algorithm for failure mode classification based on sectional area and compare results.



a. Gross section yielding



b. Net section Yielding



c. Block shear failure

[Figure 1.1 Shows failure modes in steel member image downloaded from:

<http://tudr.thapar.edu:8080/jspui/bitstream/10266/1001/3/1001.Diwaker%20Kumar%2880781008%29.pdf>]

1.3 IS-800 codal provision:

The factored design tension T , in the members shall satisfy the following requirement:

$$T < T_d$$

Where,

T_d = design strength of the member.

And the design strength of a member under axial tension, T_d , is lowest of the:

1. Design strength due to yielding of gross section, T_{dg} ,

$$T_{dg} = \frac{A_g f_y}{\gamma_{m0}}$$

Where,

f_y = yield stress,

A_g = gross area as show in fig 1.1a

γ_{m0} = partial safety factor for failure in yeilding

2. Rupture strength of critical section T_{dn} , and

$$T_{dn} = 0.9 \frac{A_n f_u}{\gamma_{m1}}$$

Where,

f_u = ultimate stress,

A_n = net area as show in fig 1.1b

γ_{m1} = partial safety factor for failure in ultimate stress

3. Block shear T_{db} ,

$$T_{dg} = \frac{A_{vg} f_y}{\sqrt{3} \gamma_{m0}} + 0.9 \frac{A_{tn} f_u}{\sqrt{3} \gamma_{m1}}$$

Or

$$T_{dg} = \frac{A_{vn} f_u}{\sqrt{3} \gamma_{m1}} + 0.9 \frac{A_{tg} f_y}{\sqrt{3} \gamma_{m0}}$$

Where,

f_y = yield stress,

f_u = ultimate stress,

γ_{m0} = partial safety factor for failure in yielding

γ_{m1} = partial safety factor for failure in ultimate stress

A_{vg} , A_{vn} = minimum gross and net area in shear along bolt line parallel to external force fig 1.1c

A_{tg} , A_{tn} = minimum gross and net area in tension along bolt line parallel to external force fig 1.1c

1.4 Organization of this report:

The IS code specifies how we can design members for axial tension, from the above equations observe some of the parameters on which design load depends are yield stress, ultimate stress, gross area, net area etc. so the mode of rupture of the section i.e., gross failure, net failure or block failure will be dependent on these parameters. The intention of this study is generating machine learning models to suggest failure mode identification for a variety of cases. Only bolted connections were included in the study, to determine the probability of these failure modes for a given section, the data has been collected from various research papers with different parameters and machine learning models are generated to train the data, and to see if there is any direct relation between these parameters and failure mode. The Data required for developing the ML is prepared from a total of 61 different values of various sections single angle, double angle, W and T sections, with different parameters such as connected and unconnected leg length and thickness, pitch, gauge, edge and end distance, no. of bolts, bolt diameter, yield and ultimate stress etc. A total of 17 parameters to train with 24 net shear section and 36 block shear sections also, a sample is collected for 40 sections with single and double angle sections whose parameters are in accordance with IS-800 such as f_y , f_u , A_{go} , A_{nc} , A_{gv} , A_{gt} , A_{nv} , A_{nt} a total of 8 parameters the A_{go} and A_{nc} parameter was used to develop model for predicting failure mode only using sectional area parameter. The report is divided in 2 parts/chapters the first part will cover basic exploratory data analysis used to engineer raw data from research papers into usable form and 2nd part will be applying the machine learning models on dataset and comparing the results from both the dataset.

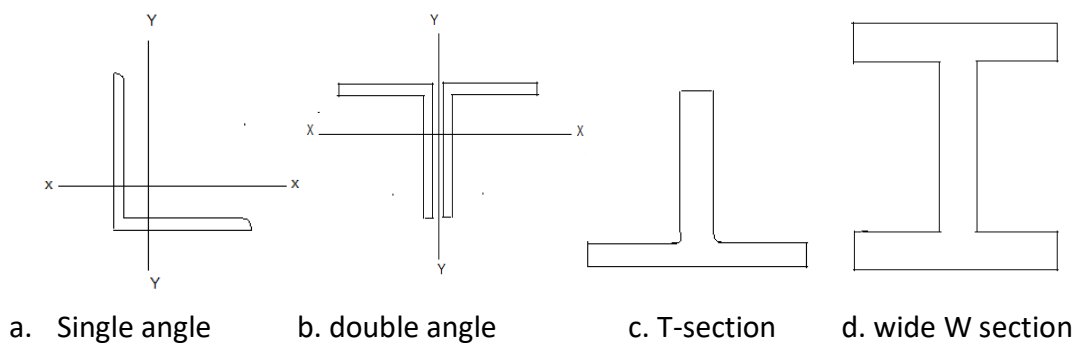


Fig 1.2 Sections used for the study

Chapter 2 exploratory data analysis

As the success of the machine learning model depends on a well-constructed database, directly applying the algorithm to the raw data might give the results but there can be huge errors or the algorithm might take longer to run. The first thing to do with the dataset is decide the important feature/parameters that have maximum importance in predicting the results, these are the 4 exploratory analyses done:

2.1 Feature selection:

These are the total 17 features of the dataset that were considered important for failure mode determination, with 4 types of sections S, D, T and W the image is showing a first few values out of the dataset, there can be 2 or more features that are similar or might almost give the same information there is no harm in keeping those features but these can increase the computation time of our algorithm as it has to go through more of these values in loops so we can remove one of these parameters, in statistics this similarity is called as correlation it measures how two variables are related to each other.

Type of section	connected leg(mm)	Unconnected leg (length in mm)	Connected leg thickness	Unconnected leg thickness	Number of pitches	Number of gauges	pitch distance	Gauge distance	Edge distance of bolt (along load direction)	Edge distance of bolt (perpendicular to load direction)	number of bolts	Diameter of bolt hole	Yield stress	Ultimate stress	ultimate load kN	failure mode 0 = net, 1 = block
S	102	102	6.4	6.4	5		76	63.5	38	38	6	24	300	450	512.8	1
S	102	102	6.4	6.4	5		76	63.5	38	38	6	24	300	450	520.8	1
S	102	102	6.4	6.4	5		76	63.5	38	38	6	24	300	450	487.1	1
S	76	76	4.8	4.8	5		76	44.5	38	38	6	24	300	450	276.9	1
S	102	102	6.4	6.4	5		76	63.5	38	38	6	24	300	450	446.4	1
S	102	102	6.4	6.4	5		76	44.5	38	38	6	24	300	450	404.9	1
S	102	102	6.4	6.4	5		76	63.5	38	38	6	24	300	450	432.9	1
S	76	54	9.5	9.5	5		76	44.5	38	38	6	24	300	450	415	1

							5						0		.2	
S	76	76	4.8	4.8	5		76	44. 5	38	38	6	24	30 0	450	233 .5	1
S	76	76	4.8	4.8	3		76	44. 5	38	38	4	24	30 0	450	239 .6	1
S	76	76	4.8	4.8	1		76	44. 5	38	38	2	24	30 0	450	198 .4	1
D	102	102	6.4	6.4	5		76	63. 5	38	38	6	24	30 0	450	973 .3	1
D	102	102	6.4	6.4	5		76	63. 5	38	38	6	24	30 0	450	997 .2	1
D	102	102	6.4	6.4	5		76	63. 5	38	38	6	24	30 0	450	990	1
D	76	76	4.8	4.8	5		76	44. 5	38	38	6	24	30 0	450	492	1
D	102	102	6.4	6.4	5		76	63. 5	38	38	6	24	30 0	450	838	1
D	102	102	6.4	6.4	5		76	63. 5	38	38	6	24	30 0	450	85	1
D	102	102	6.4	6.4	5		76	44. 5	38	38	6	24	30 0	450	797 .3	1
D	102	102	6.4	6.4	5		76	44. 5	38	38	6	24	30 0	450	782 .1	1
D	102	102	6.4	6.4	5		76	63. 5	38	38	6	24	30 0	450	857 .2	1
D	76	51	9.5	9.5	5		76	44. 5	38	38	6	24	30 0	450	814 .8	1
D	76	51	4.8	4.8	5		76	44. 5	38	38	6	24	30 0	450	412 .8	1
D	76	51	4.8	4.8	3		76	44. 5	38	38	4	24	30 0	450	439 .4	1
D	76	51	4.8	4.8	1		76	44. 5	38	38	2	24	30 0	450	344 .6	1
S	152. 4	101.6	7.82	7.82	1		76. 2	101 .6	63.5	50.8	2	27	34 6	490	362 .1	0
S	152. 4	101.6	7.52	7.52	1		76. 2	88. 9	63.5	63.5	2	27	34 6	490	444 .8	0
S	152. 4	101.6	7.67	7.67	1		76. 2	76. 2	63.5	76.2	2	27	34 6	490	500	0
T	177. 8	247.8	5.66	5.66	1	1	76. 2	12	63.5	50.8	2	27	33 5	463	264 .7	0
T	177. 8	247.8	5.54	5.54	1	1	76. 2	114 .3	63.5	63.5	2	27	33 5	463	311 .8	0
T	177. 8	247.8	5.49	5.49	1	1	76. 2	101 .6	63.5	76.2	2	27	33 5	463	345 .2	0
T	177. 8	247.8	5.66	5.66	2	1	76. 2	114 .3	63.5	63.5	3	27	33 5	463	379	0
T	177. 8	247.8	5.66	5.66	2	1	76. 2	101 .2	63.5	76.2	3	27	33 5	463	427 .5	0

T	177. 8	247.8	5.59	5.59	2	1	76. 2	88. 9	63.5	88.9	3	27	33 5	463	491 .1	0
T	177. 8	247.8	5.44	5.44	3	1	76. 2	114 .3	63.5	63.5	4	27	33 5	463	452 .8	0
T	177. 8	247.8	5.54	5.54	3	1	76. 2	101 .6	63.5	76.2	4	27	33 5	463	520 .4	0
T	177. 8	247.8	5.66	5.66	3	1	76. 2	88. 9	63.5	88.9	4	27	33 5	463	578 .2	0
W	304. 8	355.6	5.26	5.26	3	3	63. 5	152 .4	38.1	38.1	4	20. 6	37 3	510 .2	289 .1	0
W	304. 8	355.6	5.26	5.26	3	3	63. 5	152 .4	31.8	31.8	4	20. 6	38 7.5	526 .1	280 .2	0
W	304. 8	355.6	5.26	5.26	5	3	63. 5	152 .4	38.1	38.1	6	20. 6	38 7.5	526 .1	471 .9	0
W	304. 8	355.6	5.28	5.28	3	3	63. 5	76. 2	50.8	50.8	4	20. 6	38 7.5	526 .1	347 .4	0
W	304. 8	355.6	5.28	5.28	5	3	63. 5	76. 2	61	61	6	20. 6	38 7.5	526 .1	533 .8	0
T	133. 85	105.1	10.1 6	6.35	1	1	66. 67	69. 85	96.1 5	32	2	23. 81	33 2.3 2	493 .66		0
T	166	102.3	11.8 1	7.23	1	1	66. 67	88. 9	77.1	77.1	2	23. 81	33 2.3 2	493 .66		0
T	166	102.3	11.8 1	21.7	1	1	66. 67	88. 9	77.1	77.1	2	23. 81	33 2.3 2	493 .66		0
T	146. 5	131.9	11.1 7	6.6	1	1	66. 67	69. 85	76.6 5	76.65	2	23. 81	33 2.3 2	493 .66		0
T	146. 5	80.4	11.1 7	6.6	1	1	66. 67	69. 85	76.6 5	76.65	2	23. 81	33 2.3 2	493 .66		0
T	101. 7	154.4	8.89	5.96	1	1	57. 15	57. 15	89.3 5	89.35	2	20. 63	33 2.3 2	493 .66		0
S	65	85	6	6	1		50. 1	38. 5	28.7	26.5	2	23	31 0	470		0
S	65	85	6	6	1		49. 9	38. 6	28.6	26.4	2	23. 6	31 0	470		0
S	85	65	6	6	1		51	57. 8	29.4	27.2	2	23. 7	31 0	470		0
S	85	65	6	6	1		51. 3	46. 1	29.7	38.9	2	23. 8	31 0	470		0
S	85	65	6	6	1		50	57. 3	30.2	27.7	2	23. 7	31 0	470		0
S	65	85	6	6	1		49	38. 5	30.9	26.5	2	23. 7	64 0	715		0
S	65	85	6	6	1		60.	36.	29.8	28.3	2	23.	64	715		0

							1	7				9	0			
S	65	85	6	6	1		50.2	35.1	30.3	29.9	2	23.8	640	715		0
S	85	65	6	6	1		49.9	57.6	31.5	27.4	2	23.7	640	715		0
S	85	65	6	6	1		49.9	45.5	30.5	39.6	2	23.8	640	715		0
S	85	65	6	6	1		50.7	57.3	29.9	27.7	2	23.6	640	715		0
S	65	65	6	6	1		49.9	36.9	29.3	28.1	2	23.8	640	715		0
S	125	65	6	6	1		49.3	97.6	30	27.4	2	23.9	640	715		0

a. Multi-feature dataset

w/t	Fy	fu	bs/lc	Agv	Agt	Anv	Ant	Ago	Anc	Failure mode
15.9375	300	450	0.418684	2675.2	246.4	1830.4	169.6	632.32	478.72	Net
15.9375	300	450	0.418684	2675.2	246.4	1830.4	169.6	632.32	478.72	Net
15.9375	300	450	0.418684	2675.2	246.4	1830.4	169.6	632.32	478.72	Net
15.8333	300	450	0.304474	2006.4	151.2	1372.8	93.6	353.28	238.018	Net
15.9375	300	450	0.041868	2675.2	246.4	1830.4	169.6	632.32	478.72	Net
15.9375	300	450	0.368684	2675.2	368	1830.4	291.2	632.32	478.72	Net
15.9375	300	450	0.418684	2675.2	246.6	1830.4	169.6	632.32	478.72	Net
5.6842	300	450	0.234211	3971	299.25	2717	185.45	467.875	448.875	Net
15.8333	300	450	0.304474	2006.4	151.2	1372.8	93.6	353.28	238.08	Net
15.8333	300	450	0.507456	1276.8	151.2	873.6	93.6	353.28	238.08	Net
15.8333	300	450	1.522368	547.2	151.2	374.6	93.6	353.28	238.08	Net
15.975	300	450	0.418684	5350.4	492.8	3660.8	339.2	1264.64	957.44	Net
15.9375	300	450	0.418684	5350.4	492.8	3660.8	339.2	1264.64	957.44	Net
15.9375	300	450	0.418684	5350.4	492.8	3660.4	339.2	1264.64	957.44	Net

15.833 3	300	450	0.30447 4	4012.8	302.4	2745.6	187.2	706.56	476.16	Net
15.937 5	300	450	0.41868 4	5350.4	492.8	3660.8	339.2	1264.64	957.44	Net
15.937 5	300	450	0.41868 4	5350.4	492.8	3660.8	339.2	1264.64	957.44	Net
15.937 5	300	450	0.36868 4	5350.4	736	3660.8	582.4	1264.64	957.44	Net
15.937 5	300	450	0.36868 4	5350.4	736	3660.8	582.4	1264.64	957.44	Net
15.937 5	300	450	0.41868 4	5350.4	493.8	3660.8	339.2	1264.64	957.44	Net
5.3684	300	450	0.22631 6	7942	598.5	5734	370.5	878.75	957.44	Net
10.625	300	450	0.23868 4	4012.8	302.4	2745.6	187.2	466.56	476.16	Net
10.625	300	450	0.39780 7	2553.6	302.4	1747.2	187.2	466.56	476.16	Net
10.625	300	450	1.19342 1	1094.4	302.4	748.8	187.2	466.56	476.16	Net
12.992 3	346	490	2.56404 2	1092.45 4	397.25 6	775.74 4	291.668 6	763.935 8	950.051 8	Block
13.510 6	346	490	2.40131 2	1050.54 4	477.52	745.98 4	376	735.756 8	914.732 8	Block
13.246 4	346	490	2.23267 7	1071.49 9	548.45 4	760.86 4	480.909	749.857 6	932.403 6	Block
14.166 7	310	470	2.34530 9	472.8	159	265.8	90	492	234	Block
14.166 7	310	470	2.35671 3	471	158.4	258.6	87.6	492	230.4	Block
10.833 3	310	470	2.29019 6	482.4	163.2	269.1	92.1	372	349.8	Block
10.833 3	310	470	2.14873 3	486	233.4	271.8	162	372	349.2	Block
10.833 3	310	470	2.326	481.2	166.2	267.9	95.1	372	349.8	Block
14.166 7	640	715	2.39795 9	479.4	159	266.1	87.9	492	229.8	Block
14.666 7	640	715	1.92512 5	539.4	169.8	324.3	98.1	492	228.6	Block
14.166 7	640	715	2.27290 8	483	179.4	268.8	108	492	229.2	Block
10.833 3	640	715	2.33667 3	488.4	164.4	275.1	93.3	372	349.8	Block
10.833 3	640	715	2.00941 9	482.4	237	268.2	165.6	372	349.2	Block
10.833 3	640	715	2.29388 6	483.6	166.2	271.2	95.4	372	350.4	Block
10.833	640	715	1.92784	475.2	168.6	261	97.2	372	229.2	Block

3			4							
10.833	640	715	3.17647	478.8	164.4	260.7	92.7	372	588.6	Block
3			1							

b. Area features data set

Table 2.1 The prepared dataset

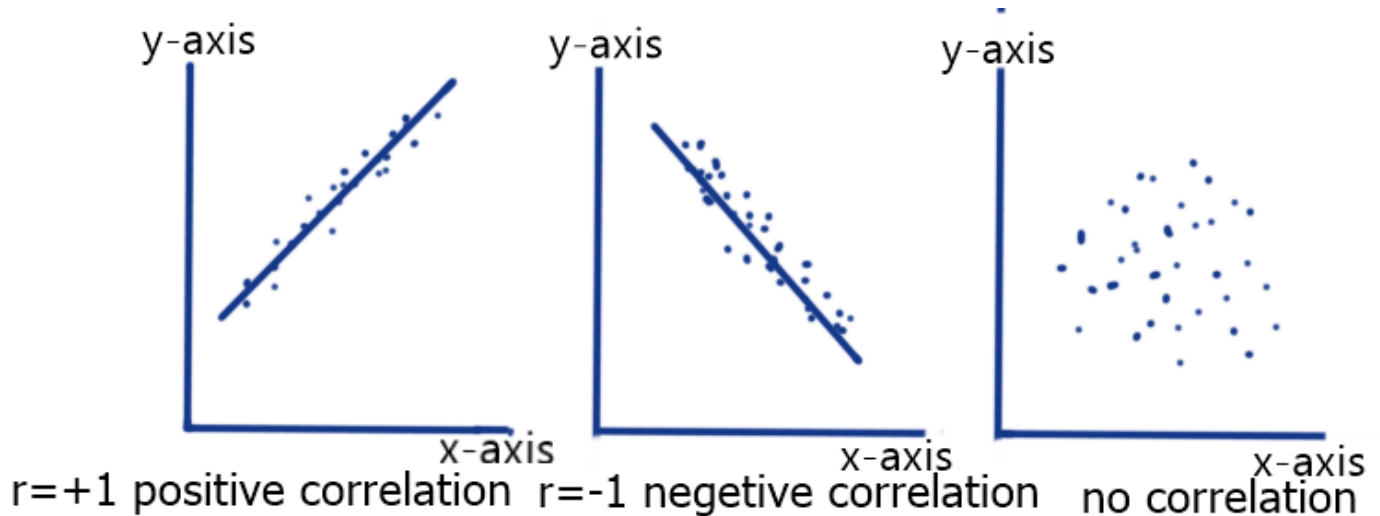


Fig 2.2 Correlation

one of the methods of finding correlation is the Pearson's coefficient of correlation the coefficient of correlation is denoted by r and $r=+1$ for positive correlation and -1 for highly negative correlation and 0 if no correlation as shown in fig 2.2, the correlation between two quantities is found by,

$$r = \frac{\sum(X - \bar{X})(Y - \bar{Y})}{\sqrt{\sum(X - \bar{X})^2 \sum(Y - \bar{Y})^2}}$$

Where,

\bar{X} is the mean of variable X and

\bar{Y} is the mean of variable Y.

If correlation values between 2 parameters is close to 1 or -1 the one of them can be removed to get higher speed of the algorithm, below is the heatmap of r values between the 17 variables in dataset.

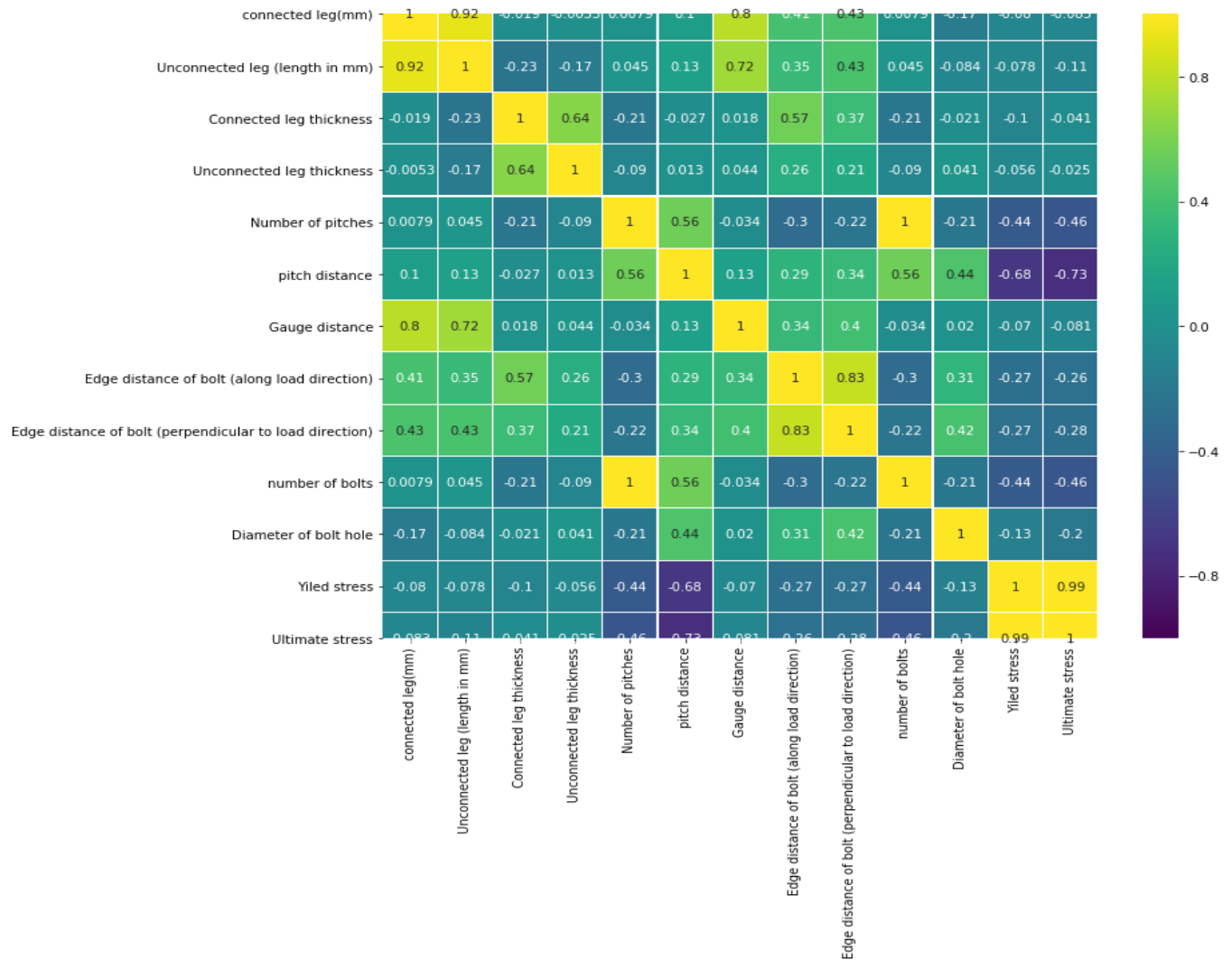


Fig 2.3 Correlation values of features

Observe, the yield stress and ultimate strength, number of bolts and number of pitches have high correlation so one of these features can be removed.

2.2 Pair-plots:

Pair-plots can be drawn between each variable in pairs to get a better look at the data distribution, for correlations and to visualize the data, the seaborn library in python allows us to draw pair-plot the below image shows the plotting between some of the features of the dataset.

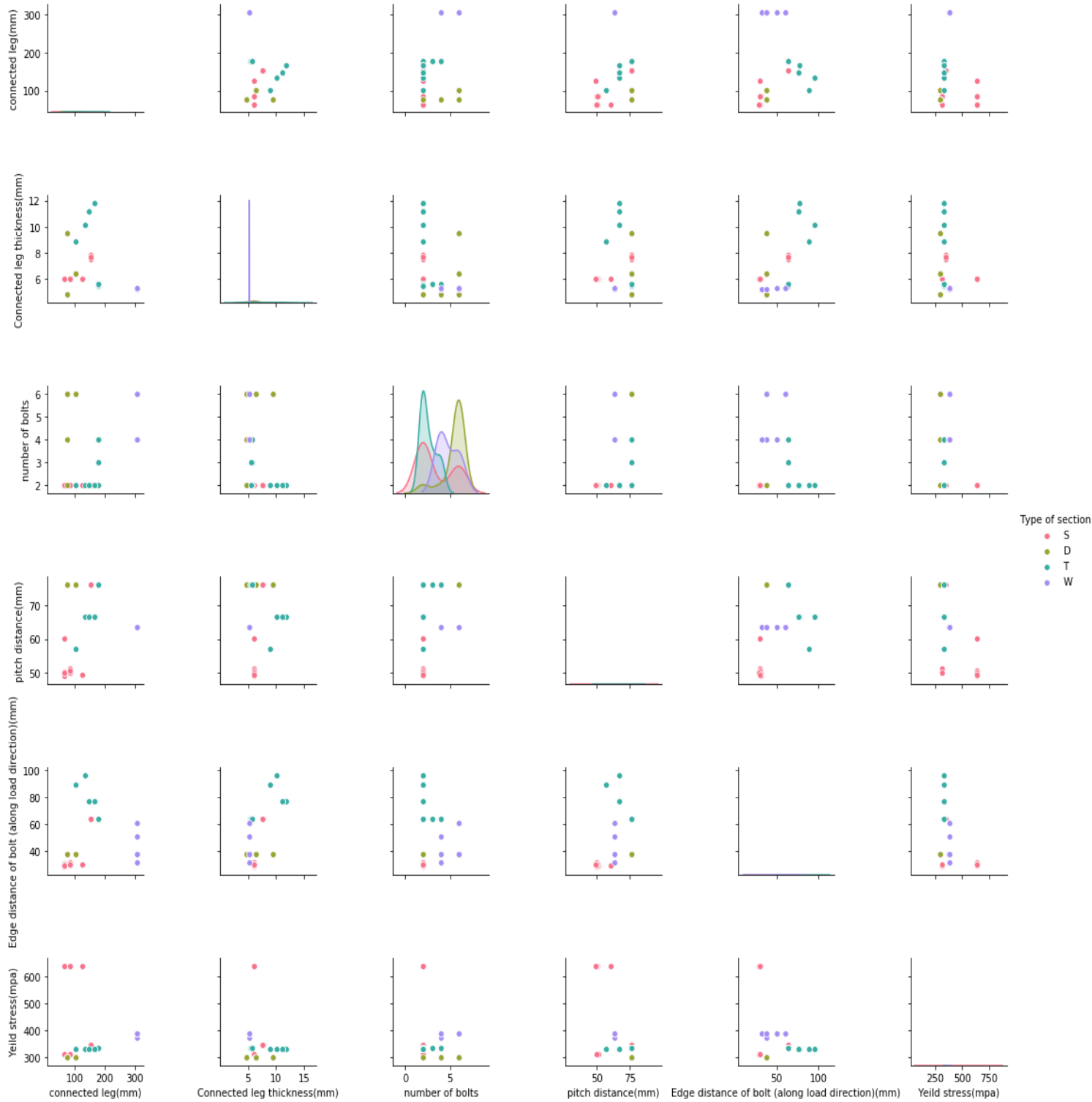


Fig 2.4 Pair-plots for 1st data-set shows there isn't seen any specific correlation of the features in dataset. A total of 60 samples with 24 net failure and 36 block-shear. A total of 13 D section, 27 S-sections, 15 T-section and 5 W-sections.

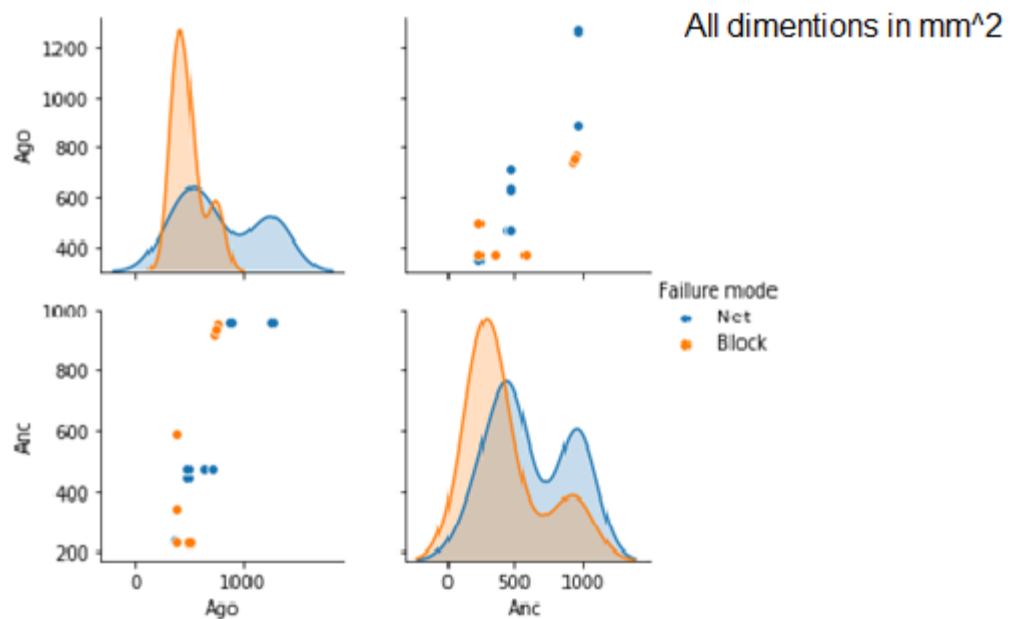


fig 2.5 Pair-plots for data set 2 of Ago and Anc with 16 block failure and 24 net failure total 40 values.

2.3 Outlier treatment:

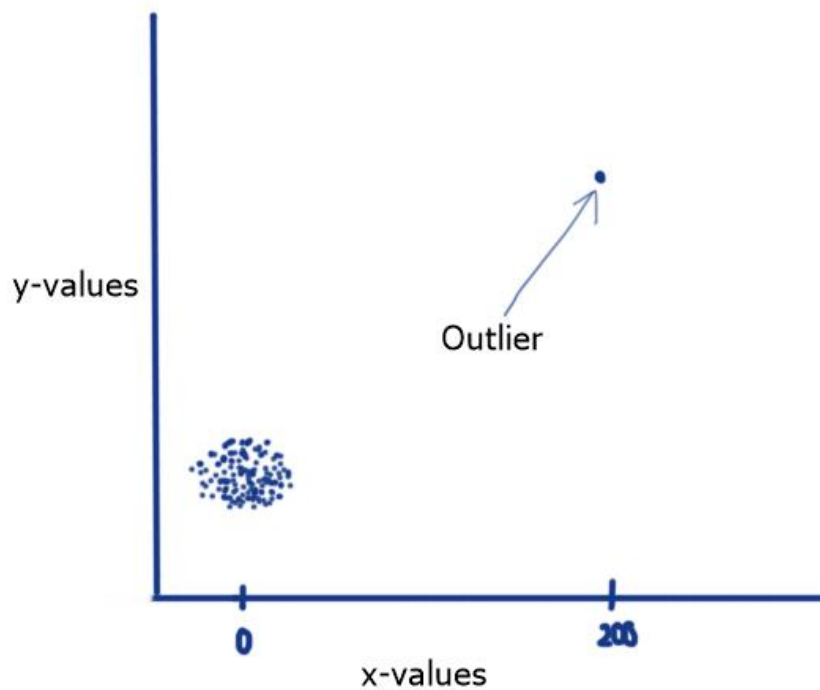


Fig 2.6 Outliers

An outlier is an observation that lies an abnormal distance from other values in a random sample from data. Outliers can cause slight errors in the results of prediction, it can be checked by z-score method where z is calculated by,

$$Z = \frac{X_n - X_{\text{mean}}}{SD_x}$$

the values which are above 3 are discarded this is around 99 percentiles of the data.

The heatmap shows the z values in the dataset, Unconnected leg seems to have an unusually large z-score to check the outlier box plot can be plotted for unconnected leg.

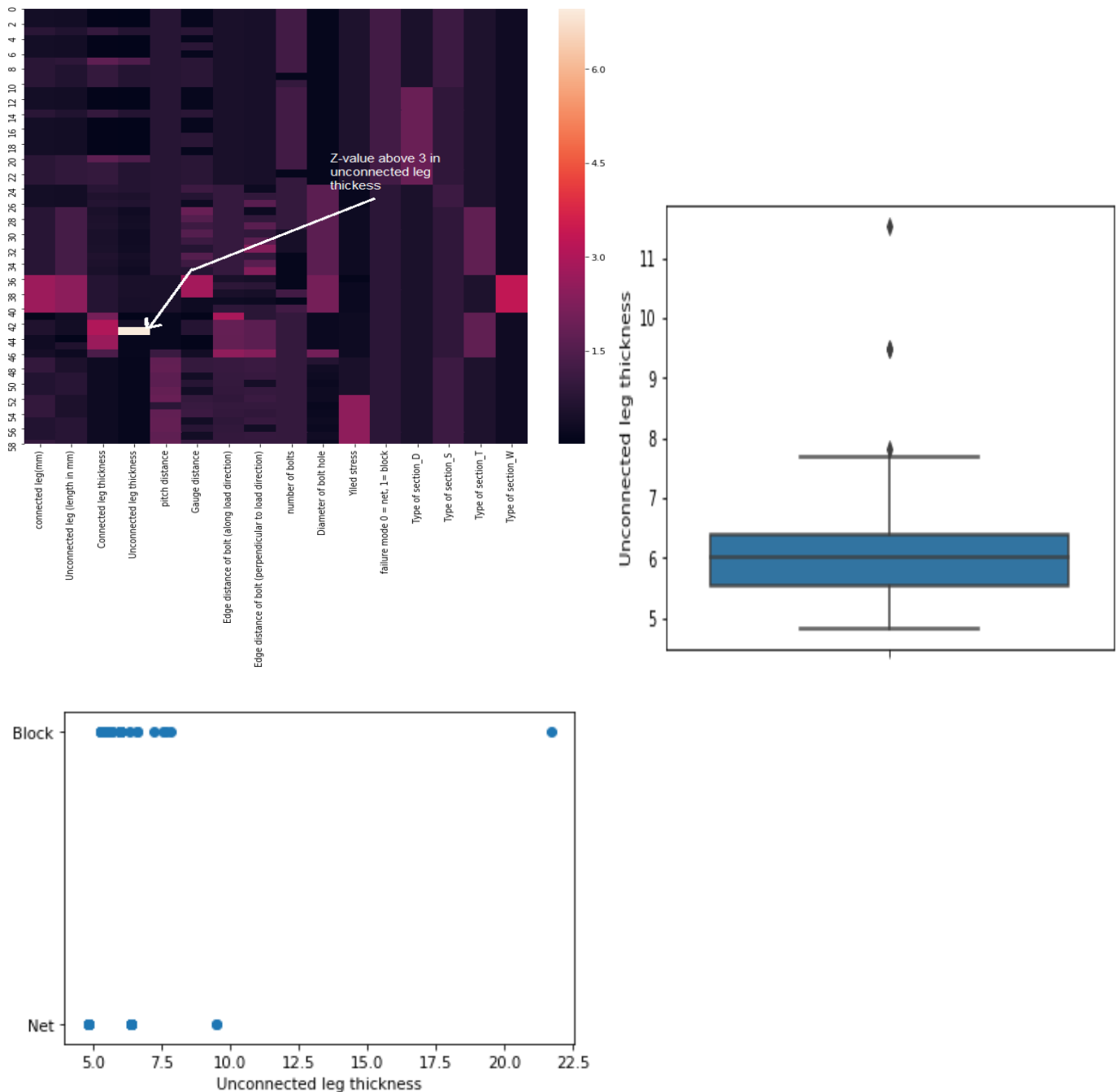


Fig 2.7 Z-score, box plot, and scatter plot of unconnected leg thickness

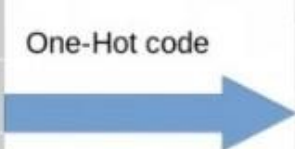
The 42nd value of the unconnected leg thickness is less than 3 times the 99 percentiles. Hence, it can be kept in the dataset.

2.3 Variable encoding and scaling the data:

It is important that there aren't any categorical variables in the dataset as the computer cannot make sense from it, such variables can be encoded with numerical values

There were 4 categorical values Single angle(S) section, double angle(D) section, T and W sections

The type of encoding used is one hot encoding this is useful here as the variables S, D, T and W do not have any order. Each category is mapped with a binary variable containing either 0 or 1. Here, 0 represents absence, and 1 represents the presence of that category. Figure shows the encoding results:



The diagram illustrates the One-Hot code transformation. On the left, a table with two columns, 'Index' and 'Animal', shows four rows: Index 0 with 'S', Index 1 with 'D', Index 2 with 'T', and Index 3 with 'W'. A large blue arrow labeled 'One-Hot code' points from this table to a second table on the right. The second table has five columns: 'Index', 'S', 'D', 'T', and 'W'. It shows the binary encoding for each category: Index 0 has S=1, D=0, T=0, W=0; Index 1 has S=0, D=1, T=0, W=0; Index 2 has S=0, D=0, T=1, W=0; and Index 3 has S=0, D=0, T=0, W=0.

Index	Animal
0	S
1	D
2	T
3	W

Index	S	D	T	W
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	0

Fig 2.8 Variable encodings of the sections

Also, before applying the algorithm it is also important to check the scale of data, some values can be very high and some can be close to 0 due to unit conversions to fix that we normalize the datapoints using this formula on each of the feature.

$$X' = \frac{X - \mu}{\max(x) - \min(x)}$$

Where,

X' is mean normalized value

X is original value

μ is the mean of the feature

after performing these initial data analyses, predicting model can be developed,

Chapter 3 Building the model:

These machine learning algorithms were used to develop predicting models and their results are as show:

3.1 Logistic regression:

Logistic regression is a linear model it finds a linear relationship between the features and the output (failure mode), for the dataset we have to predict 1 if failure mode is net section and predict 0 if the failure mode is block shear.

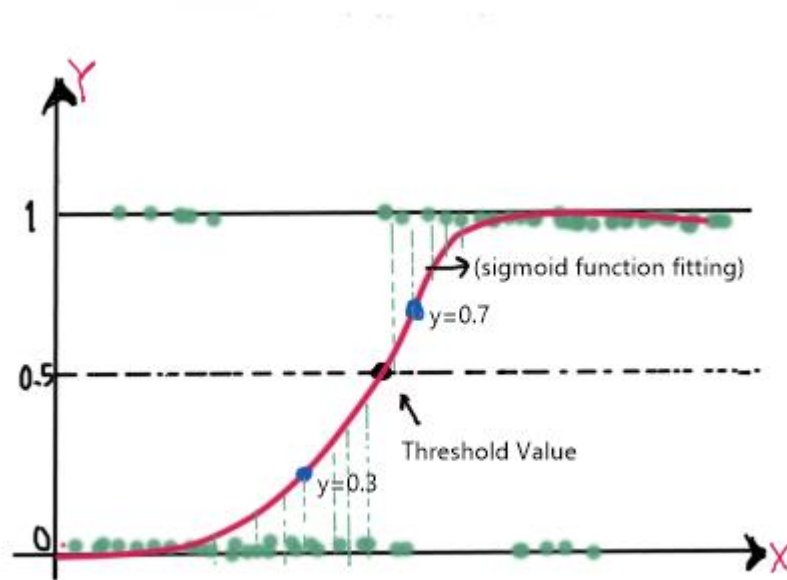


Fig 3.1 Sigmoid function

Consider the linear combinations of the failure mode parameters as Z ,

$$Z_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \dots + \beta_n x_n = \sum_{i=1}^m \beta_0 + \beta_i x_i$$

where, $\beta_1, \beta_2, \beta_3, \dots, \beta_n$ are random constants and $x_1, x_2, x_3, \dots, x_n$ are the features such as connected leg length thickness, pitch, gauge length, bolt diameter etc.

we only want the output in between 0 and 1, so applying a function that gives output in range 0-1 hence apply sigmoid function to Z which returns values in range 0-1 given as,

$$y_{predicted}(x_i) = \frac{1}{1+e^{-(Z_i)}} = \frac{1}{1+e^{-(\sum_{i=1}^m \beta_0 + \beta_i x_i)}}$$

$y_{predicted}(x)$ gives the predicted values of failure mode from our features X_i 's between 0(net section) and 1(block shear). only need is to adjust the β_i 's values for right predictions. Initialize β_i 's with random values. So, the error in real values and predicted values can be calculated as,

$$error(x_i) = \frac{\sum_{i=1}^m (y_{predicted}(x_i) - y_{actual}(x_i))^2}{m} \text{ known as mean squared error}$$

Where, m is number of data and y_{actual} , $y_{predicted}$ is the given failure mode and predicted failure

The aim of prediction is minimizing error with respect to β_i 's so now the problem is of minimization and finding the value of β_i 's at which the error is minimum this is done through numerical method of gradient descent,

The error Vs β_i graphs is somewhat shown below, to reach the minimum error start from random β and update β as given below with each iteration.

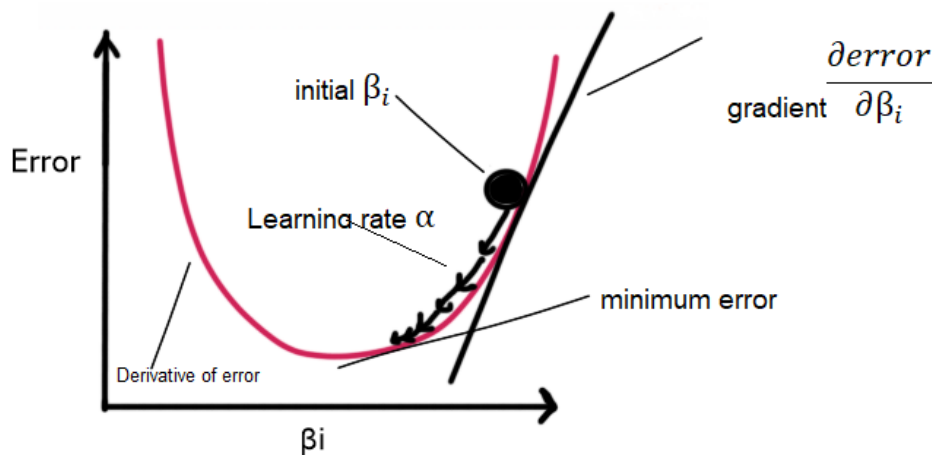


Fig 3.2 Gradient descent

Repeat

{

$$\beta_i = \beta_i - \alpha \frac{\partial error}{\partial \beta_i} \text{ for all } \beta_i \text{'s}$$

$$\text{or } \beta_i = \beta_i - 2\alpha \frac{\sum_{i=1}^m (y_{\text{predicted}}(x_i) - y_{\text{actual}}(x_i))}{m}$$

}

Where α is learning rate telling how fast or slow gradient descent steps

The gradient decent tunes in the value of β_i according to the data inputted in and gives us right predictions.

Starting from random value of β_i the algorithm will reach minimum when $\frac{\partial error}{\partial \beta_i}$ is 0 at that point there

will be no change in β_i value and $y_{\text{predicted}}(x)$ at those values of β_i will be the failure mode probability

The results of this algorithm are summarised below:

When using only area features of the section i.e., with Ago(gross area of outstanding leg) and Anc(net area of connected leg) the distribution of the dataset is as follows,

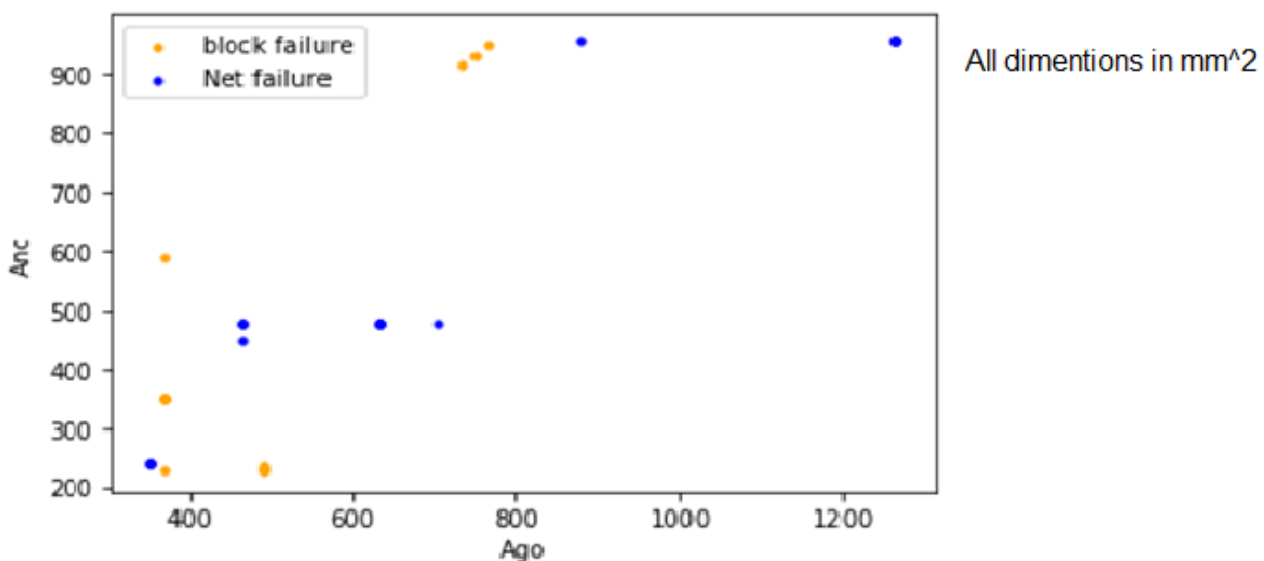


fig 3.3 Ago Vs Anc

The logistic regression will create a decision boundary to classify the points as blue(block) or orange(net) the regression line that model develops is given by,

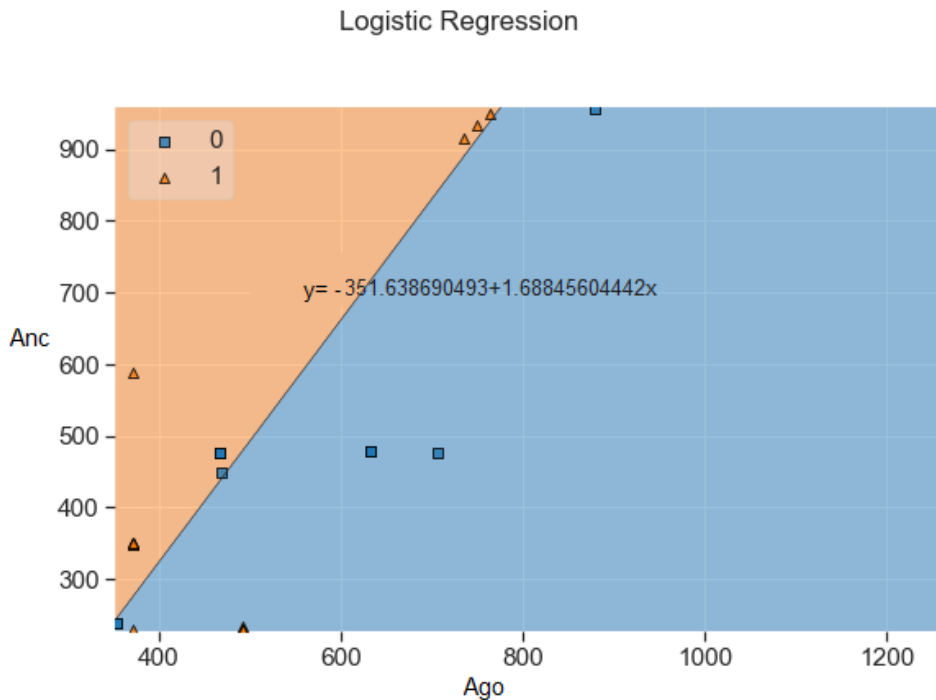


Fig 3.4 Decision boundary of linear logistic regression

The points to the left are orange/1 or net failure and points to right is blue/0 or block failure as seen in the plot there are some points miss classified as well, as logistic regression can only draw linear boundary hence, there are errors the percentages of these errors are shown in the form of the confusion-matrix,

		Predicted class		
		Positive	Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error	Sensitivity $\frac{TP}{(TP + FN)}$
	Negative	False Positive (FP) Type I Error	True Negative (TN)	Specificity $\frac{TN}{(TN + FP)}$
		Precision $\frac{TP}{(TP + FP)}$	Negative Predictive Value $\frac{TN}{(TN + FN)}$	Accuracy $\frac{TP + TN}{(TP + TN + FP + FN)}$

Fig 3.5 Confusion matrix

True positives are the no. of values which are 1(block failure) and correctly predicted as 1 by the algorithm and the true negatives are 0 (net failure) and correctly predicted as 0. False positives are negating or 0 but predicted wrong as 1 this is type 1 error and false negatives are value 1 predicted wrongly as 0 this is

type 2 error the percentages of errors and accuracy measures are calculated by the precision, specificity, sensitivity and accuracy as shown in the fig 3.5 And the confusion matrix of the linear logistic regression is shown below,

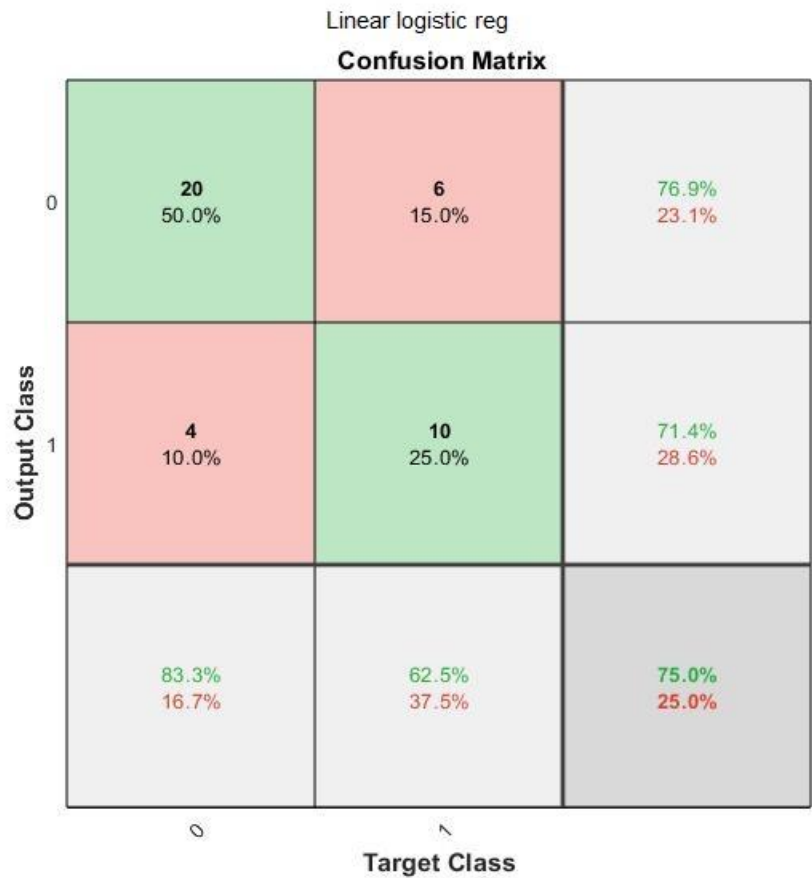
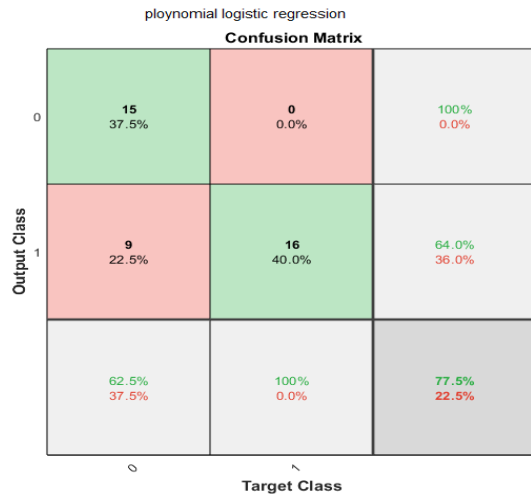
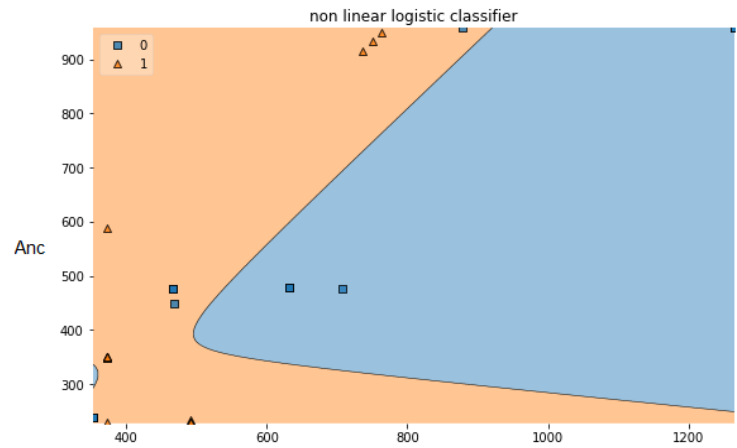


Fig 3.6 Confusion matrix of single feature linear logistic regression

When a non-linear quadratic feature combination in logistic regression was implemented the confusion matrix and decision boundary is as shown,



a. Confusion matrix

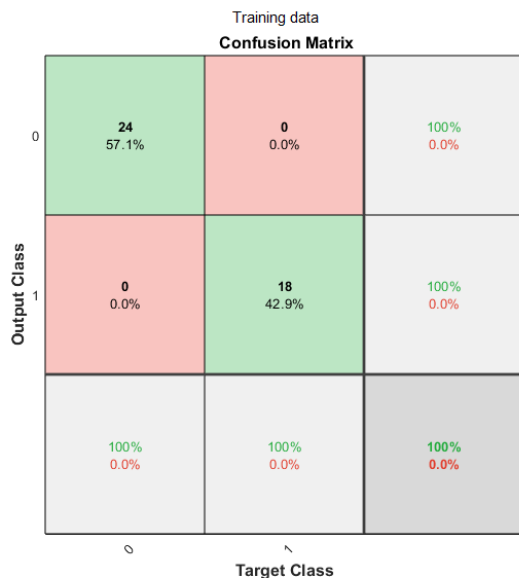


b. Quadratic decision boundary

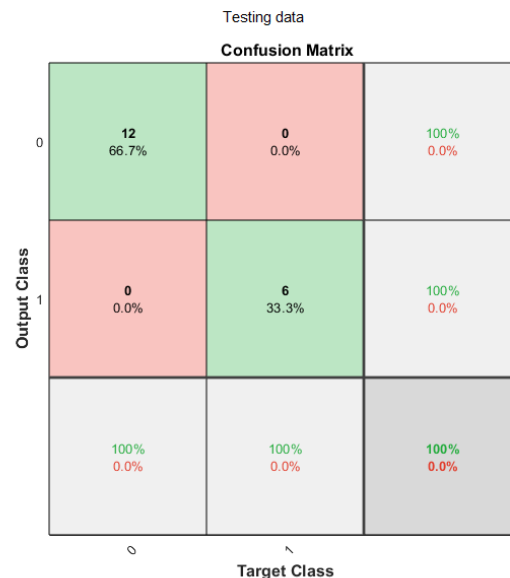
Fig 3.7 Single feature non-linear logistic regression

When using only the area factors to predict the over-all accuracy of logistic regression remains 75% for linear and 77.5% for polynomial classifier and rest points are wrongly classified out of 40, there are 9-10 wrongly classified points.

When the same logistic regression classifier was applied to the other dataset with multiple features the dataset had 60 points with 70% (42) for training and rest 30% (18) points for testing the accuracy was 100% in both train and test and all points were classified as shown in below confusion plot.



a. Training data

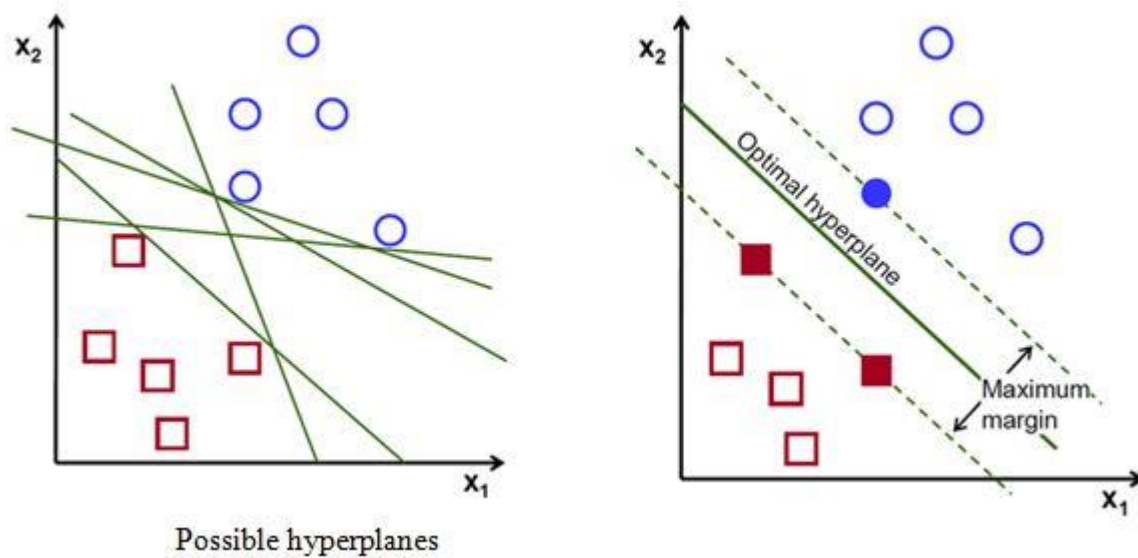


b. testing data

fig 3.8 multi-feature logistic classifier

3.2 Support vector machine:

Support vector fits the data linearly as well as non-linearly this algorithm is also called large marginal classifier as it classifies the dataset into large marginal distance between positives and negatives i.e., there will be large margin gap between net section and block shear datapoints. The reason to choose svm is because it gives high margin for new data points which can be helpful depending on the data.



[Fig 3.9 Large margin classification decision boundary with largest margin, source:

<https://towardsdatascience.com/svm-feature-selection-and-kernels-840781cc1a6c>]

The algorithm for Svm is also similar to logistic regression only the way of calculating error is different,

The linear combination of the features can be defined again as,

$$y_{\text{predict}} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \dots + \beta_n x_n = \sum_{i=1}^m \beta_0 + \beta_i x_i$$

Now instead of calculating mean square error we use hinge loss error given by,

$$\text{error}(x) = \sum_{i=1}^m \max(0, 1 - y_{\text{actual}} y_{\text{predicted}}) = \sum_{i=1}^m \max(0, 1 - y_i (\beta_i x_i + \beta_0))$$

$$\begin{aligned} \text{error} &= 0, & \text{if } y_{\text{actual}}y_{\text{predicted}} > 1 \text{ else} \\ &= 1 - y_{\text{actual}}y_{\text{predicted}} \end{aligned}$$

This error function ensures that the $y_{\text{predicted}}$ values are in range of 0-1 and also the margin of classification to be wider.

Now, the aim of prediction is minimizing error with respect to β_i 's so now the problem is of minimization and finding the value of β_i 's at which the error is minimum this is done through numerical method of gradient descent,

Repeat

{

$$\beta_i = \beta_i - \alpha \frac{\partial \text{error}}{\partial \beta_i} \text{ for all } \beta_i \text{'s}$$

}

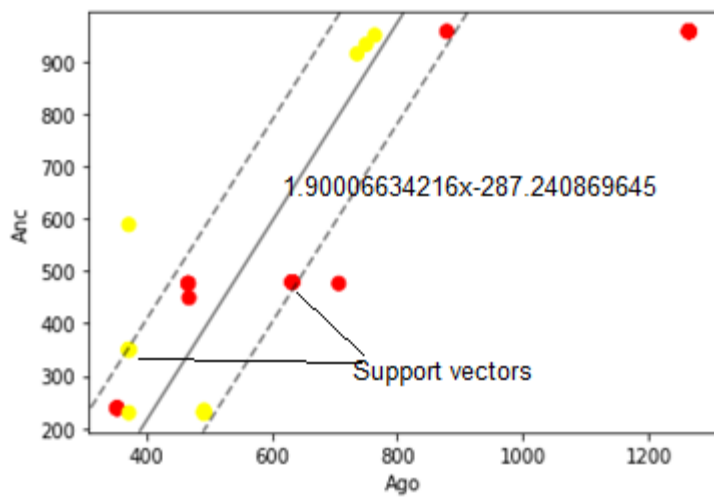
Where α is learning rate telling how fast or slow gradient descent steps

The gradient decent tunes in the value of β_i according to the data inputted in and gives us right predictions.

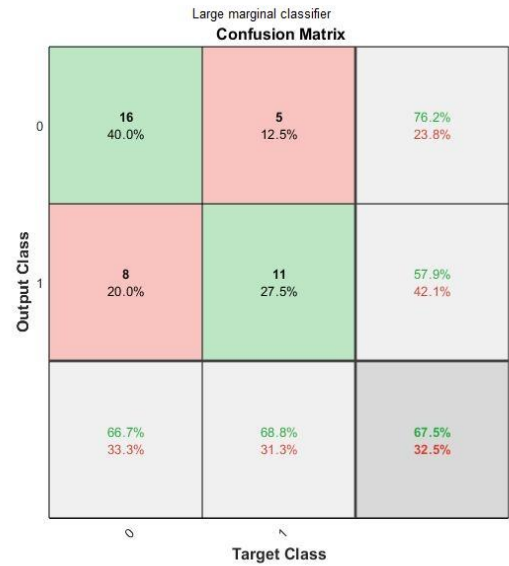
Starting from random value of β_i the algorithm will reach minimum when $\frac{\partial \text{error}}{\partial \beta_i}$ is 0 at that point there

will be no change in β_i value and $y_{\text{predicted}}(x)$ at those values of β_i will be the failure mode probability

The results of this algorithm are shown below:

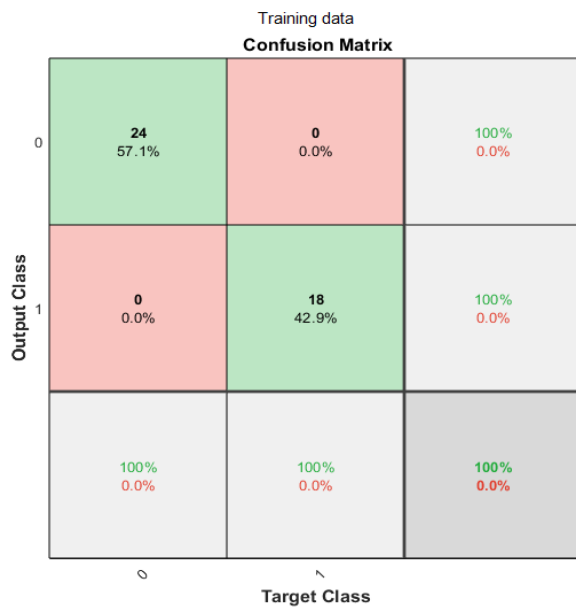


a. Decision boundary

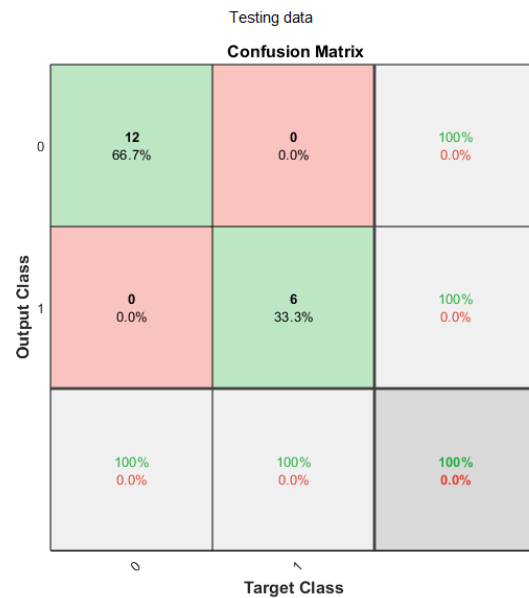


b. confusion matrix

Fig 3.10 Svm for single feature



a. Training data



b. testing data

fig 3.11 multi-feature svm classifier

the accuracy of single feature svm classifier is less than logistic regression at 67.5% this is because it maximized the margin between classes and the multi-feature classifier has accuracy of 100%.

3.3 Decision trees and random forest:

Decision tree is a tree-based algorithm they are a non-linear decision maker with linear decision surfaces in other words they can adapt to any kind of problem in hand decision tree is one single decision tree but random forest creates many trees with random subparts of dataset to form a random forest and does the classification by majority voting. Random forest algorithms have very high accuracy and result can be interpreted as a tree after classification is performed. The reason to choose DT and RF is that DT gives results that can be drawn in form of tables and RF have very high accuracy.

A decision tree consists of the root /Internal node which further splits into decision nodes/branches, depending on the outcome of the branches the next branch or the terminal /leaf nodes are formed.

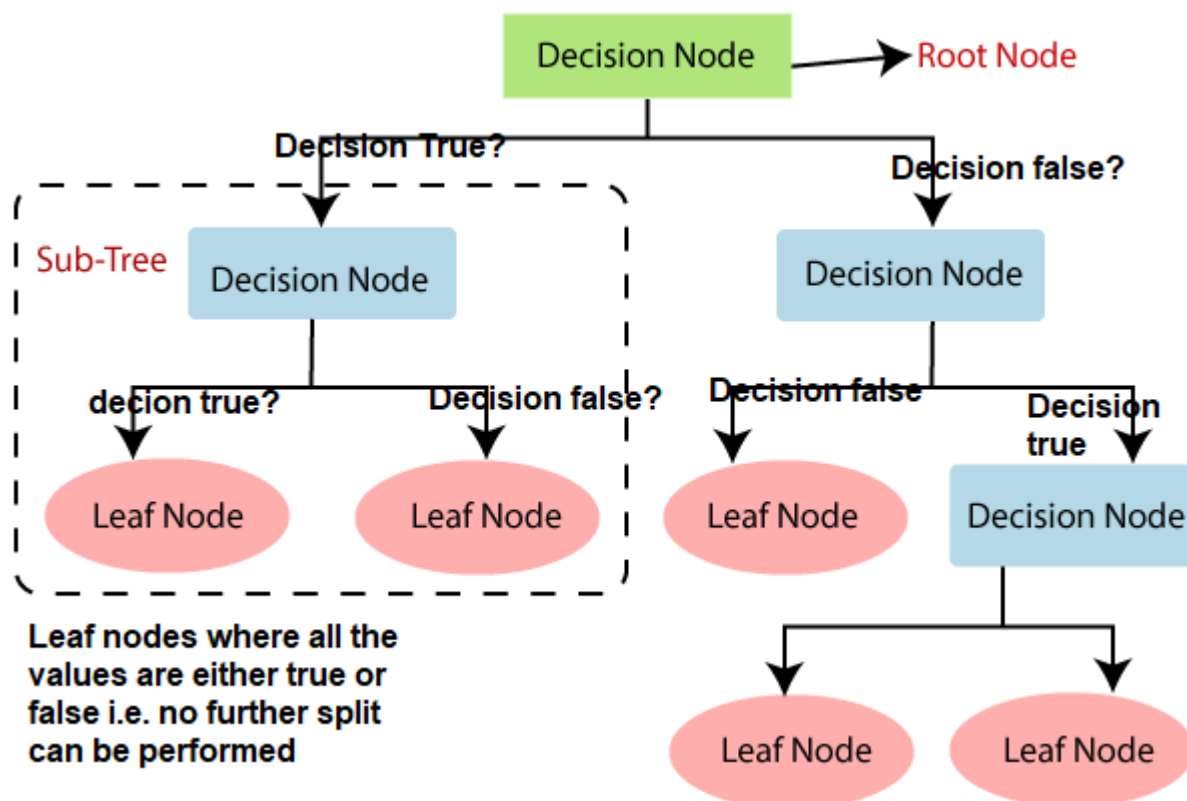


Fig 3.12 Decision tree structure

An intuition of if else can be thought of as an If else statement it will test the attribute and if it holds true than it will go to left node or else right node and split will happen until there are only true 0 nodes or true 1 nodes left in a node called leaf node.

Creating a decision tree is just a matter of choosing which attribute should be tested at each node in the tree.

information gain is the measure which will be used to decide which attribute/feature should be tested at each node.

Information gain or **IG** is a statistical property that measures how well a given attribute separates the training examples according to their target classification. Constructing a decision tree is all about finding an attribute that returns the highest information gain and the smallest entropy. The information gain is calculated using the measure called entropy that measures the randomness in the information processed

Mathematically the entropy can be calculated by,

$$Entropy(S) = \frac{p}{p+n} \log_2 \frac{p}{p+n} + \frac{n}{p+n} \log_2 \frac{n}{p+n}$$

Where,

S is Current state or dataset, and

P is no. of positive samples i.e., no. of block failure

N is no. of negative samples i.e, no. of net section failure

From this entropy of the whole data-set we calculate the entropy and gain of each attribute the attribute with the highest gain will be the root node

$$gain = Entropy(S) - \frac{p_i + n_i}{p + n} Entropy(A)$$

Where,

A is attribute or features we have

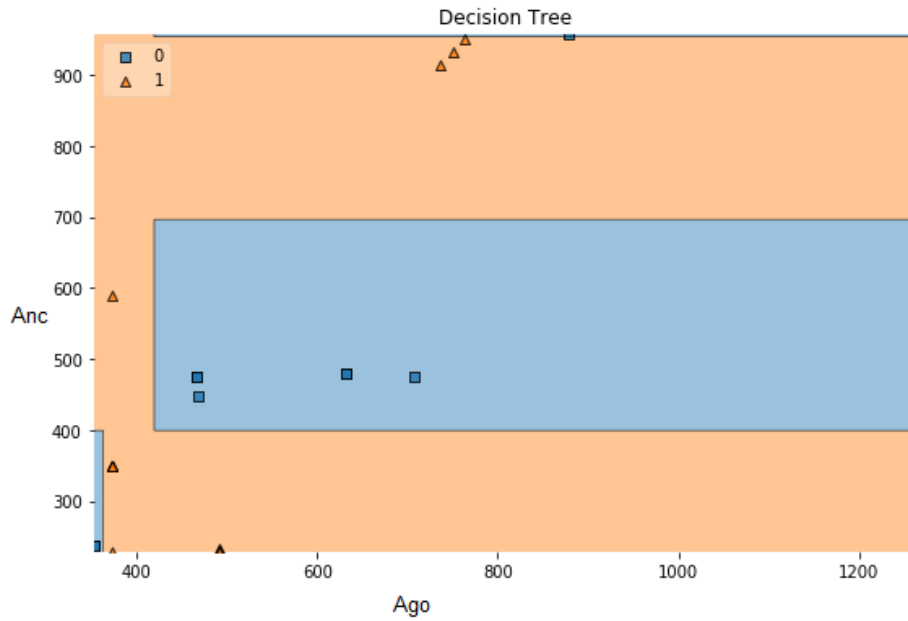
And p_i and n_i are no of positives and negative samples in the attribute A.

We have to do it for all the attributes/features A in the dataset and select the attribute with highest gain that will be the root node and if the other nodes are not leaf nodes than continue the process again with other features we have until leaf node is reached at that node gain is 0.

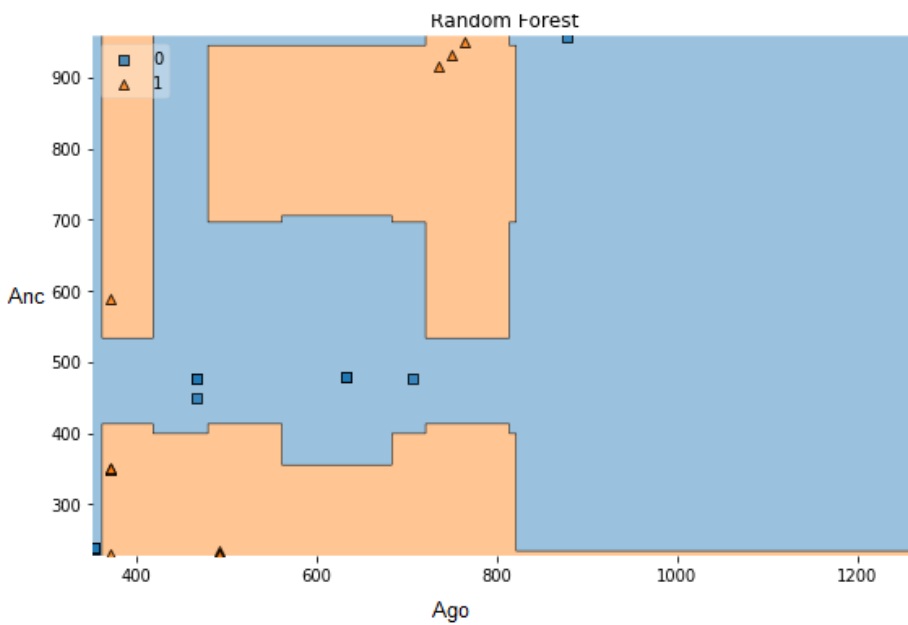
The results of the decision tree and random forest are shown below:

The random forest and decision trees have the highest accuracy they can adapt to any kind of decision boundaries the accuracy these at 100% in both single feature as well as the multiple feature classification

The decision boundary developed by decision tree and random forest is shown below for single feature classification problem:



a. decision tree boundary



b. random forest decision

Fig 3.13 Decision boundary of single feature decision tree and random forest

Both of these algorithms create a linear decision boundary but non-linear surface random forest creates an island within clusters of data points whereas, in decision tree the boundary is less complex than RF-classifier.

The tree representation of the decision surface is as shown in fig 3.14

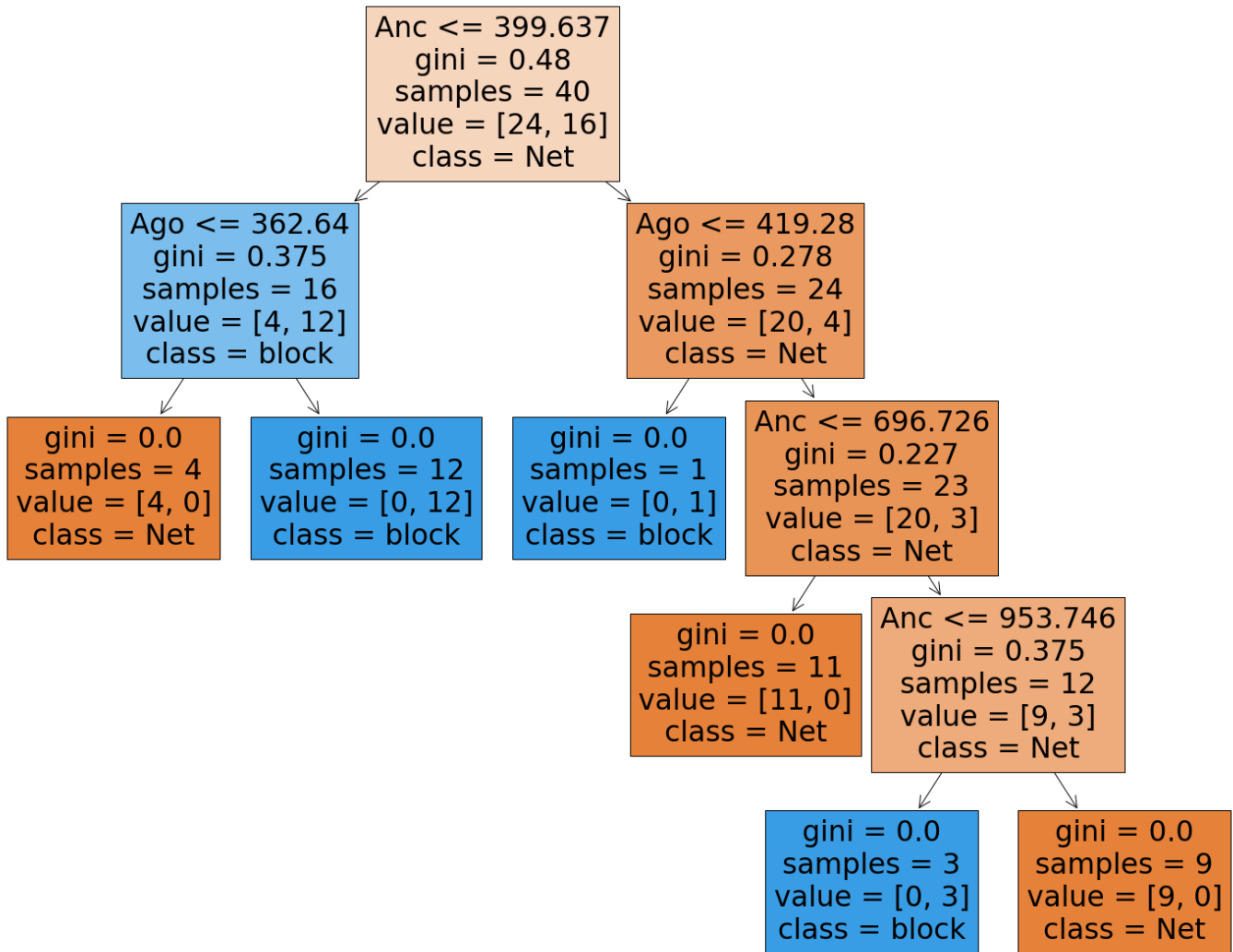
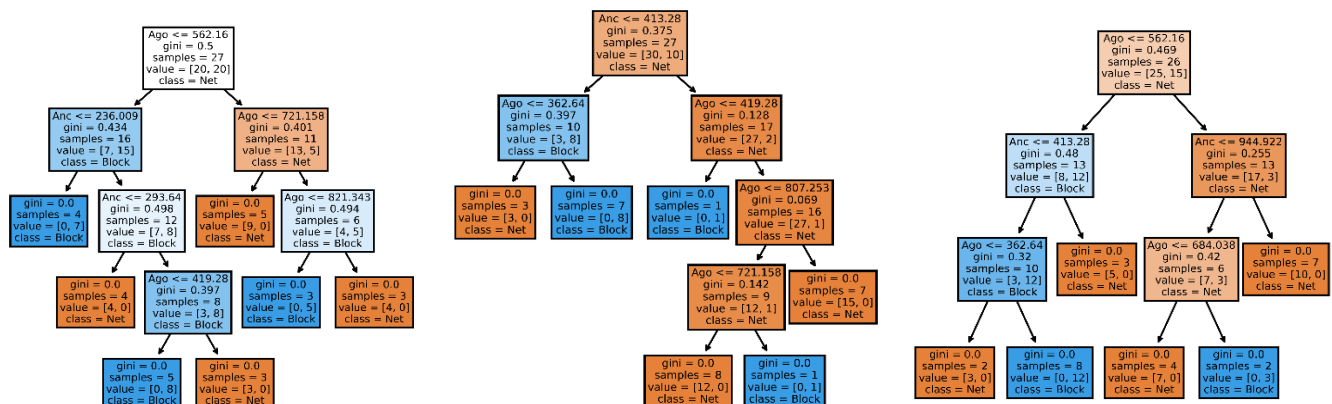


Fig 3.14 decision tree representation

As see above gini or information gain is highest for root node and the left nodes have 0 information as all the values are of only one class.

Decision tree applies to whole data set whereas, random forest uses decision tree with small subset of data a total of 9 decision trees are made out of the 9 random subparts of dataset as shown below and the classification is made by taking the majority vote.



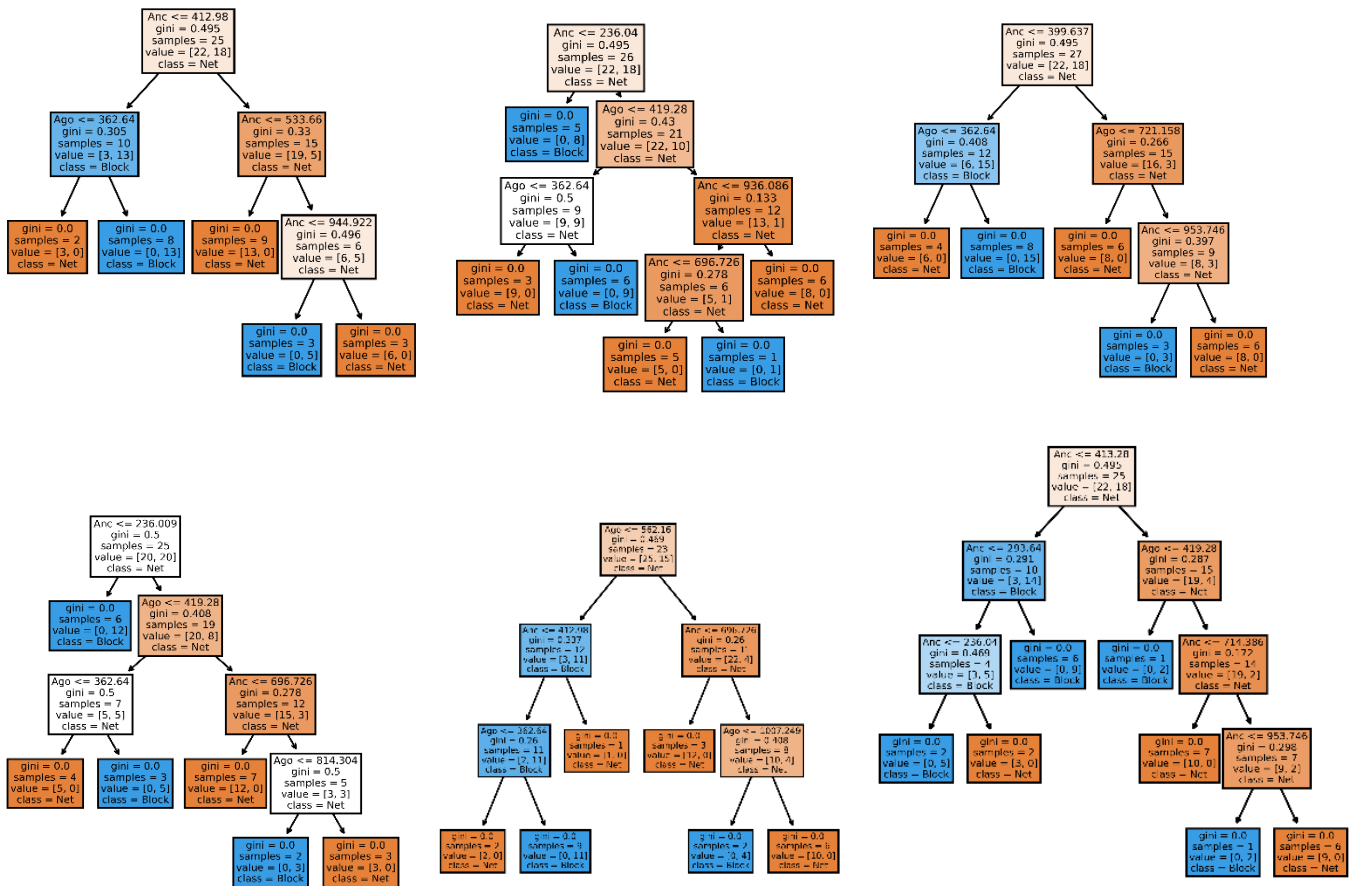


Fig 3.15 random forest trees

3.3 k-nearest neighbors and naïve bayes:

Beside these, 2 very simple models to test the results were also used.

The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm. The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. It assumes that similar data points are together KNN captures the idea of similarity (sometimes called distance, proximity, or closeness)

For each of the point in dataset measure k(usually taken 3)-closest points using Euclidian distance and take the majority vote out of these closest neighbors. In this way a decision boundary is created.



[Fig 3.16 K-nearest neighbor classifier source: <https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn>]

A Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The crux of the classifier is based on the Bayes theorem in probability.

Bayes theorem,

$$P\left(\frac{A}{B}\right) = P\left(\frac{B}{A}\right) * \frac{P(A)}{P(B)}$$

Using bayes we can determine probability of A, given B has already occurred.

In the dataset A is probability of 0(net failure) or 1(block failure) given the feature vector B i.e., with given features what is the probability that the failure mode will be 0 or 1.

Naïve bayes uses a naïve assumption that the features are independent i.e., all the predictors' features have an equal effect on the probability. Hence, it's called naïve bayes but this assumption works fine in many cases.

For the dataset bayes theorem can be written as,

$$P\left(\frac{y}{X}\right) = \frac{P\left(\frac{X}{y}\right)P(y)}{P(X)}$$

X is given as $(x_1, x_2, x_3 \dots x_n)$ which are the parameters of failure modes.

And $y=0$ or 1 net or block

applying the independent event assumption to the feature X the probability can be written as,

$$P\left(\frac{y}{x_1, x_2, x_3 \dots x_n}\right) = \frac{P\left(\frac{x_1}{y}\right)P\left(\frac{x_2}{y}\right) \dots P\left(\frac{x_n}{y}\right)}{P(x_1)P(x_2) \dots P(x_n)}P(y)$$

The denominator term is always constant probability of features so therefore,

$$P\left(\frac{y}{x_1, x_2, x_3 \dots x_n}\right) \propto P(y) \prod_{i=1}^n P\left(\frac{x_i}{y}\right)$$

In the data set $P\left(\frac{y}{x_1, x_2, x_3 \dots x_n}\right)$ is either 0 or 1 therefore we can write,

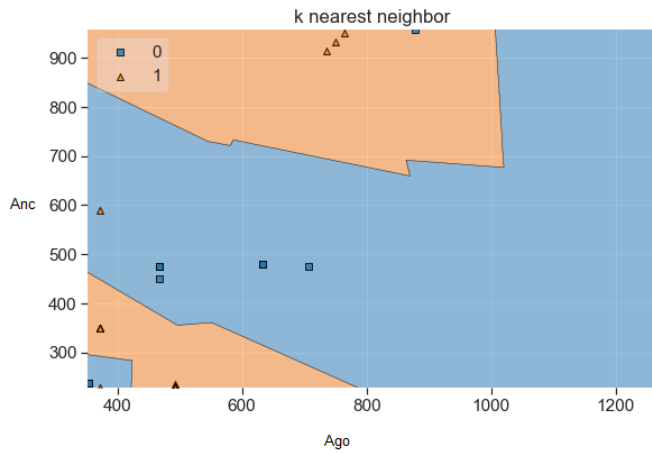
$$y = \operatorname{argmax}(P(y) \prod_{i=1}^n P\left(\frac{x_i}{y}\right))$$

$P(y)$ is the frequency of a given failure mode and $P\left(\frac{x_i}{y}\right)$ can be calculated by the gaussian distribution,

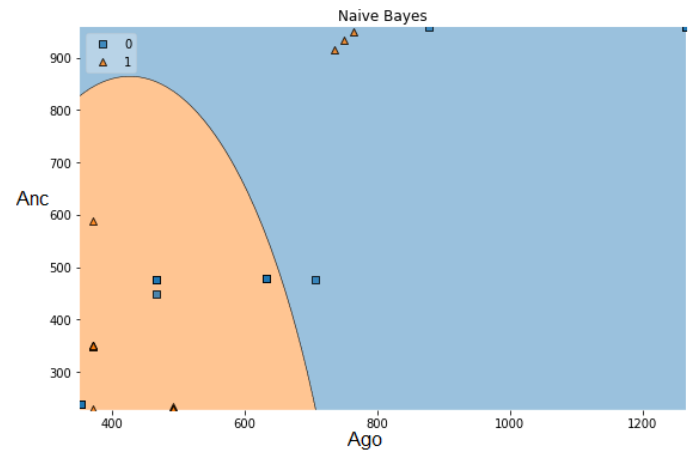
$$P\left(\frac{x_i}{y}\right) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

where, σ_y and μ_y are the mean and standard deviation of the class.

The reason to choose naïve bayes and knn is to test the accuracy of the results with other algorithms as well



a. knn



b. naïve bayes

3.17 Decision boundary for knn and naïve bayes single feature classifier

KNN

Confusion Matrix

Output Class \ Target Class	0	1	
0	23 57.5%	2 5.0%	92.0% 8.0%
1	1 2.5%	14 35.0%	93.3% 6.7%
	95.8% 4.2%	87.5% 12.5%	92.5% 7.5%

a. Knn

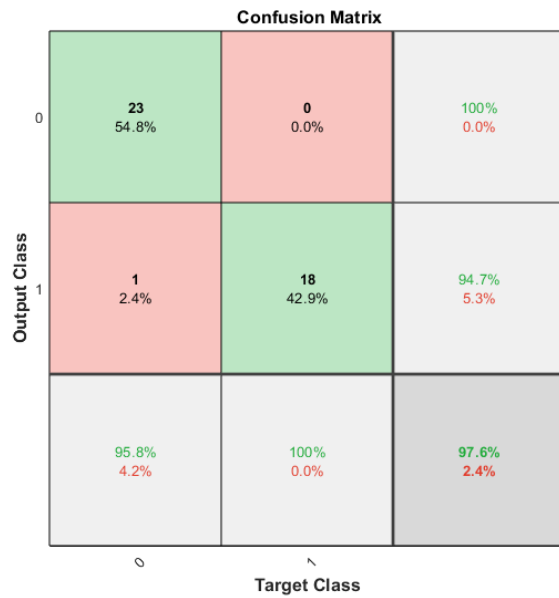
Naive bayes

Confusion Matrix

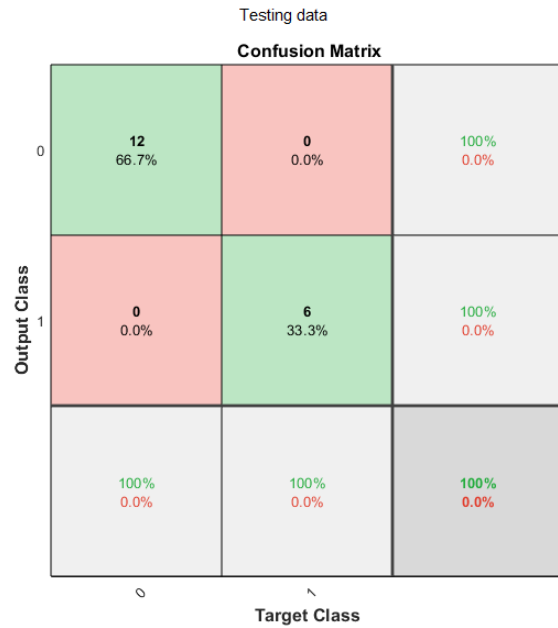
Output Class \ Target Class	0	1	
0	10 25.0%	3 7.5%	76.9% 23.1%
1	14 35.0%	13 32.5%	48.1% 51.9%
	41.7% 58.3%	81.3% 18.8%	57.5% 42.5%

b. naïve bayes

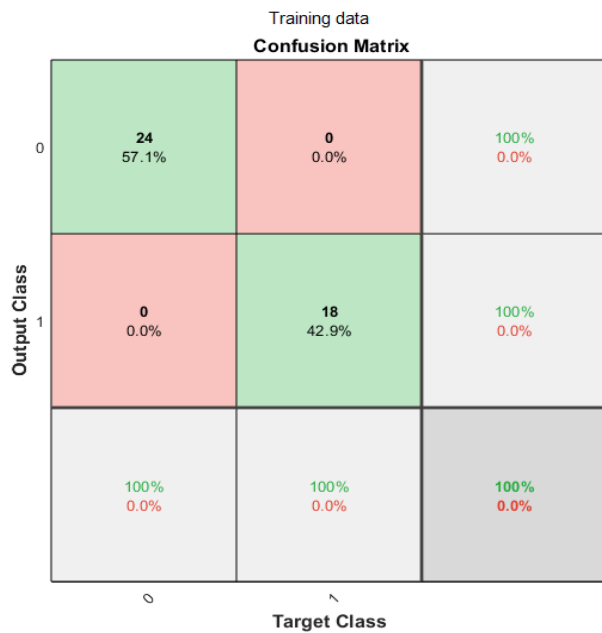
fig 3.18 single-feature knn and naïve bayes classifier



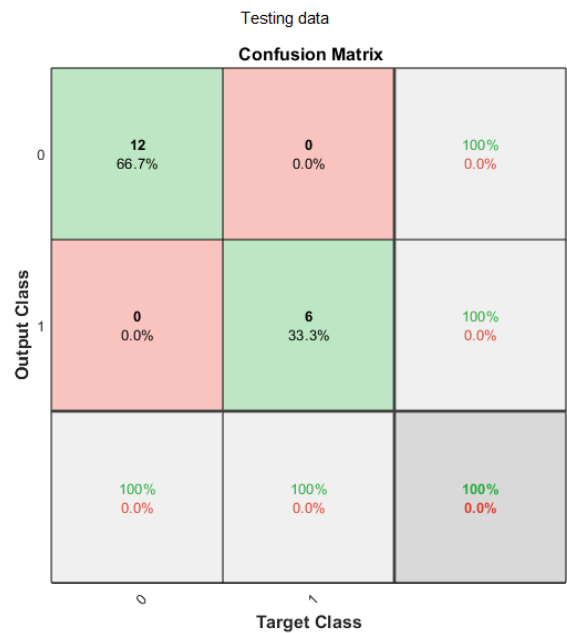
a. knn Training data



b. knn testing data



4 Naïve bayes training data



b. naïve bayes testing data

fig 3.19 multi-feature knn and naïve bayes classifier

the knn is dividing a non-linear decision boundary with accuracy of 92.5% for single feature dataset and 97.6% for multi-feature dataset whereas naïve bayes has an accuracy of 57.5% for single feature and 100% for multi-feature.

Chapter 5

Conclusion and Future work:

5.1 Conclusion

This study aimed at the estimation of the failure mode probabilities in steel tension had two parts Firstly, developing the machine learning models for multiple features (connected leg length, thickness, pitch, gauge, f_y , f_u , bolt details etc.) and comparing the accuracy of various different models developed. Second, using only the sectional area parameters (Ago, Anc etc. as specified by the IS-800 code) to develop a smart algorithm for failure mode classification on the basis of section area only and compare results. The observation made was the average accuracy for first part was around 99% with multiple parameters data is well fit and whereas, the accuracy with only using the sectional area parameter was around 81.5%. the model cannot fit single parameter accurately but still advance algorithms such as random forest still had 100% in single parameter classification. The data-based model can be a faster way to determine failure mode for member design, the precision and accuracy still needs to be tested with large chunks of data as more and more study is done in this and data is generated the model will keep improving, the study has shown the classification can be done with data-based method as well with good accuracy of prediction.

5.2 Future work

With the amount of data generated every day, combined with the advanced computing power that the computers have today, and the advance algorithms that learn hidden patterns and relations in data, these techniques can be further applied to engineering structures, tension members considering the welded connection and also bolted connection parameters, as well as developing the model with larger datasets with varying parameters deep learning models for higher accuracy with large datasets can be developed, there have not been much study about this as data available is scarce, but this study has shown that predicting task can be learnt by a computer as well. So, as more data is generated every day the model will keep improving itself and have higher accuracy.

References

[1] <https://www.journals.elsevier.com/engineering-structures>

[2] Geoffrey L. Kulak, Nd Eric Yue Wu,- journal of construction steel research “shear lag in blot angle tension member, vol. 558., pp. 1148–1146.

[3] James G Orbison, mark E Wagner, William P fritz,- journal of construction steel research 49 (1999) 225-239 “Tenison plane behavior in single row bolted connections subject to block shear” vol. 558., pp. 229.

<https://www.udemy.com/course/data science a to z/>

<https://web.stanford.edu/~hastie/ElemStatLearn/>

<https://www.coursera.org/learn/machine-learning>

[4] H.I. Epstein, m j McGinnis,- computers and structures 77(2000) 571-582 “finite element modelling of structural Tees” pp 574.

[5] Indian standard-800 “General construction in steel -code of practice,” New Delhi: Bureau of Indian Standards, 2007.

[6] <https://towardsdatascience.com/>

[7] Sujith manglathu, hansol jang, seong hoon hwang, jong-su jeon,- engineering structures 208 (2020) 11031 “Data driven machine learning-based seismic failure mode identification of reinforced concrete shear walls”

