%%-\*- text -\*-

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% This is a PROMISE Software Engineering Repository data set made publicly

% available in order to encourage repeatable, verifiable, refutable, and/or

% improvable predictive models of software engineering.

%

% If you publish material based on PROMISE data sets then, please

% follow the acknowledgment guidelines posted on the PROMISE repository

% web page http://promise.site.uottawa.ca/SERepository .

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% 1. Title/Topic: cocomonasa/software cost estimation

% 2. Sources:

%

% -- Creators: 60 NASA projects from different centers

% for projects from the 1980s and 1990s. Collected by

% Jairus Hihn, JPL, NASA, Manager SQIP Measurement &

% Benchmarking Element

% Phone (818) 354-1248 (Jairus.M.Hihn@jpl.nasa.gov)

%

% -- Donor: Tim Menzies (tim@barmag.net)

%

% -- Date: December 2 2004

% 3. Past Usage

% 1. "Validation Methods for Calibrating Software Effort

% Models", T. Menzies and D. Port and Z. Chen and

% J. Hihn and S. Stukes, Proceedings ICSE 2005,

% http://menzies.us/pdf/04coconut.pdf

% -- Results

% -- Given background knowledge on 60 prior projects,

% a new cost model can be tuned to local data using

% as little as 20 new projects.

% -- A very simple calibration method (COCONUT) can

% achieve PRED(30)=7% or PRED(20)=50% (after 20 projects).

% These are results seen in 30 repeats of an incremental

% cross-validation study.

% -- Two cost models are compared; one based on just

% lines of code and one using over a dozen "effort

% multipliers". Just using lines of code loses 10 to 20

% PRED(N) points.

%

% 3.1 Additional Usage:

% 2. "Feature Subset Selection Can Improve Software Cost Estimation Accuracy"

% Zhihao Chen, Tim Menzies, Dan Port and Barry Boehm

% Proceedings PROMISE Workshop 2005,

% http://www.etechstyle.com/chen/papers/05fsscocomo.pdf

% P02, P03, P04 are used in this paper.

% -- Results

% -- To the best of our knowledge, this is the first report

% of applying feature subset selection (FSS)

% to software effort data.

%

% -- FSS can dramatically improve cost estimation.

%

% ---T-tests are applied to the results to demonstrate

% that always in our data sets, removing

% attributes improves performance without increasing the

% variance in model behavior.

%

% 4. Relevant Information

%

% The COCOMO software cost model measures effort in calendar months

% of 152 hours (and includes development and management hours).

% COCOMO assumes that the effort grows more than linearly on

% software size; i.e. months=a\* KSLOC^b\*c. Here, "a" and "b" are

% domain-specific parameters; "KSLOC" is estimated directly or

% computed from a function point analysis; and "c" is the product

% of over a dozen "effort multipliers". I.e.

%

% months=a\*(KSLOC^b)\*(EM1\* EM2 \* EM3 \* ...)

%

% The effort multipliers are as follows:

%

% increase | acap | analysts capability

% these to | pcap | programmers capability

% decrease | aexp | application experience

% effort | modp | modern programing practices

% | tool | use of software tools

% | vexp | virtual machine experience

% | lexp | language experience

% ----------+------+---------------------------

% | sced | schedule constraint

% ----------+------+---------------------------

% decrease | stor | main memory constraint

% these to | data | data base size

% decrease | time | time constraint for cpu

% effort | turn | turnaround time

% | virt | machine volatility

% | cplx | process complexity

% | rely | required software reliability

%

% In COCOMO I, the exponent on KSLOC was a single value ranging from

% 1.05 to 1.2. In COCOMO II, the exponent "b" was divided into a

% constant, plus the sum of five "scale factors" which modeled

% issues such as ``have we built this kind of system before?''. The

% COCOMO~II effort multipliers are similar but COCOMO~II dropped one

% of the effort multiplier parameters; renamed some others; and

% added a few more (for "required level of reuse", "multiple-site

% development", and "schedule pressure").

%

% The effort multipliers fall into three groups: those that are

% positively correlated to more effort; those that are

% negatively correlated to more effort; and a third group

% containing just schedule information. In COCOMO~I, "sced" has a

% U-shaped correlation to effort; i.e. giving programmers either

% too much or too little time to develop a system can be

% detrimental.

%

% The numeric values of the effort multipliers are:

%

% very very extra productivity

% low low nominal high high high range

% ---------------------------------------------------------------------

% acap 1.46 1.19 1.00 0.86 0.71 2.06

% pcap 1.42. 1.17 1.00 0.86 0.70 1.67

% aexp 1.29 1.13 1.00 0.91 0.82 1.57

% modp 1.24. 1.10 1.00 0.91 0.82 1.34

% tool 1.24 1.10 1.00 0.91 0.83 1.49

% vexp 1.21 1.10 1.00 0.90 1.34

% lexp 1.14 1.07 1.00 0.95 1.20

% sced 1.23 1.08 1.00 1.04 1.10 e

% stor 1.00 1.06 1.21 1.56 -1.21

% data 0.94 1.00 1.08 1.16 -1.23

% time 1.00 1.11 1.30 1.66 -1.30

% turn 0.87 1.00 1.07 1.15 -1.32

% virt 0.87 1.00 1.15 1.30 -1.49

% rely 0.75 0.88 1.00 1.15 1.40 -1.87

% cplx 0.70 0.85 1.00 1.15 1.30 1.65 -2.36

%

% These were learnt by Barry Boehm after a regression analysis of the

% projects in the COCOMO I data set.

% @Book{boehm81,

% Author = "B. Boehm",

% Title = "Software Engineering Economics",

% Publisher = "Prentice Hall",

% Year = 1981}

%

% The last column of the above table shows max(E)/min(EM) and shows

% the overall effect of a single effort multiplier. For example,

% increasing "acap" (analyst experience) from very low to very

% high will most decrease effort while increasing "rely"

% (required reliability) from very low to very high will most

% increase effort.

%

% There is much more to COCOMO that the above description. The

% COCOMO~II text is over 500 pages long and offers

% all the details needed to implement data capture and analysis of

% COCOMO in an industrial context.

% @Book{boehm00b,

% Author = "Barry Boehm and Ellis Horowitz and Ray Madachy and

% Donald Reifer and Bradford K. Clark and Bert Steece

% and A. Winsor Brown and Sunita Chulani and Chris Abts",

% Title = "Software Cost Estimation with Cocomo II",

% Publisher = "Prentice Hall",

% Year = 2000,

% ibsn = "0130266922"}

%

% Included in that book is not just an effort model but other

% models for schedule, risk, use of COTS, etc. However, most

% (?all) of the validation work on COCOMO has focused on the effort

% model.

% @article{chulani99,

% author = "S. Chulani and B. Boehm and B. Steece",

% title = "Bayesian Analysis of Empirical Software Engineering

% Cost Models",

% journal = "IEEE Transaction on Software Engineering",

% volume = 25,

% number = 4,

% month = "July/August",

% year = "1999"}

%

% The value of an effort predictor can be reported many ways

% including MMRE and PRED(N).MMRE and PRED are computed from the

% relative error, or RE, which is the relative size of the

% difference between the actual and estimated value:

%

% RE.i = (estimate.i - actual.i) / (actual.i)

%

% Given a data set of of size "D", a "Train"ing set of size

% "(X=|Train|) <= D", and a "test" set of size "T=D-|Train|", then

% the mean magnitude of the relative error, or MMRE, is the

% percentage of the absolute values of the relative errors,

% averaged over the "T" items in the "Test" set; i.e.

%

% MRE.i = abs(RE.i)

% MMRE.i = 100/T\*( MRE.1 + MRE.2 + ... + MRE.T)

%

% PRED(N) reports the average percentage of estimates that were

% within N% of the actual values:

%

% count=0

% for(i=1;i<=T;i++) do if (MRE.i <= N/100) then count++ fi done

% PRED(N) = 100/T \* sum

%

% For example, e.g. PRED(30)=50% means that half the estimates are

% within 30% of the actual. Shepperd and Schofield comment that

% "MMRE is fairly conservative with a bias against overestimates

% while Pred(25) will identify those prediction systems that are

% generally accurate but occasionally wildly inaccurate".

% @article{shepperd97,

% author="M. Shepperd and C. Schofield",

% title="Estimating Software Project Effort Using Analogies",

% journal="IEEE Transactions on Software Engineering",

% volume=23,

% number=12,

% month="November",

% year=1997,

% note="Available from

% \url{http://www.utdallas.edu/~rbanker/SE\_XII.pdf}"}

%

% 4.1 Further classification of the projects

%

% 4.1.1 Classify the projects into different project categories - P02, P03, P04.

% (The criteria is unknown and they are disjoint.)

%

% Category sequence Original sequence\_of\_NASA

% P01 1 NASA 26

% P01 2 NASA 27

% P01 3 NASA 28

% P01 4 NASA 29

% P01 5 NASA 30

% P01 6 NASA 31

% P01 7 NASA 32

% P02 1 NASA 4

% P02 2 NASA 5

% P02 3 NASA 6

% P02 4 NASA 7

% P02 5 NASA 8

% P02 6 NASA 9

% P02 7 NASA 10

% P02 8 NASA 11

% P02 9 NASA 12

% P02 10 NASA 13

% P02 11 NASA 14

% P02 12 NASA 15

% P02 13 NASA 16

% P02 14 NASA 17

% P02 15 NASA 18

% P02 16 NASA 19

% P02 17 NASA 20

% P02 18 NASA 21

% P02 19 NASA 22

% P02 20 NASA 23

% P02 21 NASA 24

% P02 22 NASA 25

% P03 1 NASA 34

% P03 2 NASA 35

% P03 3 NASA 36

% P03 4 NASA 37

% P03 5 NASA 38

% P03 6 NASA 39

% P03 7 NASA 40

% P03 8 NASA 41

% P03 9 NASA 42

% P03 10 NASA 43

% P03 11 NASA 44

% P03 12 NASA 45

% P04 1 NASA 47

% P04 2 NASA 48

% P04 3 NASA 49

% P04 4 NASA 50

% P04 5 NASA 51

% P04 6 NASA 52

% P04 7 NASA 53

% P04 8 NASA 54

% P04 9 NASA 55

% P04 10 NASA 56

% P04 11 NASA 57

% P04 12 NASA 58

% P04 13 NASA 59

% P04 14 NASA 60

%

% 4.1.2 Classify the projects into different task categories - T01, T02, T03.

% (The criteria is unknown and they are disjoint.)

% T01:sequencing T02:avionics T03:missionPlanning

%

% Category sequence Original sequence\_of\_NASA

% T01 1 NASA 43

% T01 2 NASA 41

% T01 3 NASA 37

% T01 4 NASA 34

% T01 5 NASA 40

% T01 6 NASA 38

% T01 7 NASA 39

% T01 8 NASA 36

% T02 1 NASA 4

% T02 2 NASA 6

% T02 3 NASA 26

% T02 4 NASA 27

% T02 5 NASA 33

% T02 6 NASA 32

% T02 7 NASA 29

% T02 8 NASA 30

% T02 9 NASA 28

% T02 10 NASA 7

% T02 11 NASA 9

% T02 12 NASA 10

% T02 13 NASA 55

% T02 14 NASA 31

% T03 1 NASA 51

% T03 2 NASA 52

% T03 3 NASA 16

% T03 4 NASA 17

% T03 5 NASA 8

% T03 6 NASA 50

% T03 7 NASA 53

% T03 8 NASA 45

% T03 9 NASA 48

% T03 10 NASA 47

%

% 4.1.3 Classify the projects into different Centers - C01, C02, C03.

% (The criteria is unknown and they are disjoint.)

% Category sequence Original sequence\_of\_NASA

%

% C01 1 NASA 1

% C01 2 NASA 2

% C01 3 NASA 51

% C01 4 NASA 52

% C01 5 NASA 50

% C01 6 NASA 53

% C01 7 NASA 48

% C01 8 NASA 47

% C01 9 NASA 58

% C01 10 NASA 59

% C01 11 NASA 60

% C01 12 NASA 49

% C01 13 NASA 54

% C02 1 NASA 45

% C02 2 NASA 43

% C02 3 NASA 41

% C02 4 NASA 35

% C02 5 NASA 34

% C02 6 NASA 40

% C02 7 NASA 38

% C02 8 NASA 39

% C02 9 NASA 36

% C02 10 NASA 37

% C02 11 NASA 42

% C02 12 NASA 44

% C03 1 NASA 4

% C03 2 NASA 6

% C03 3 NASA 26

% C03 4 NASA 27

% C03 5 NASA 33

% C03 6 NASA 32

% C03 7 NASA 29

% C03 8 NASA 30

% C03 9 NASA 28

% C03 10 NASA 7

% C03 11 NASA 9

% C03 12 NASA 10

% C03 13 NASA 31

% C03 14 NASA 21

% C03 15 NASA 14

% C03 16 NASA 22

% C03 17 NASA 3

% C03 18 NASA 19

% C03 19 NASA 16

% C03 20 NASA 17

% C03 21 NASA 8

% C03 22 NASA 23

% C03 23 NASA 20

% C03 24 NASA 24

% C03 25 NASA 12

% C03 26 NASA 5

% C03 27 NASA 13

% C03 28 NASA 25

% C03 29 NASA 15

% C03 30 NASA 18

% C03 31 NASA 11

% 5. Number of instances: 60

% 6. Number of attributes: 17 (15 discrete in the range Very\_Low to

% Extra\_High; one lines of code measure, and one goal field

% being the actual effort in person months).

% 7. Attribute information:

@relation cocomonasa.csv

@attribute RELY {Nominal,Very\_High,High,Low} %1

@attribute DATA {High,Low,Nominal,Very\_High} %2

@attribute CPLX {Very\_High,High,Nominal,Extra\_High,Low} %3

@attribute TIME {Nominal,Very\_High,High,Extra\_High} %4

@attribute STOR {Nominal,Very\_High,High,Extra\_High} %5

@attribute VIRT {Low,Nominal,High} %6

@attribute TURN {Nominal,High,Low} %7

@attribute ACAP {High,Very\_High,Nominal} %8

@attribute AEXP {Nominal,Very\_High,High} %9

@attribute PCAP {Very\_High,High,Nominal} %10

@attribute VEXP {Low,Nominal,High} %11

@attribute LEXP {Nominal,High,Very\_Low,Low} %12

@attribute MODP {High,Nominal,Very\_High,Low} %13

@attribute TOOL {Nominal,High,Very\_High,Very\_Low,Low} %14

@attribute SCED {Low,Nominal,High} %15

@attribute LOC numeric %16

@attribute ACT\_EFFORT numeric %17

% 8. Missing attributes: none

% 9: Class distribution: the class value (ACT\_EFFORT) is continuous.

% After sorting all the instances on ACT\_EFFORT, the following

% distribution was found:

% Instances Range

% --------- --------------

% 1..10 8.4 .. 42

% 11..20 48 .. 68

% 21..30 70 .. 117.6

% 31..40 120 .. 300

% 41..50 324 .. 571

% 51..60 750 .. 3240

% Change log:

% -----------

%

% 2005/04/04 Jelber Sayyad Shirabad (PROMISE Librarian) <promise@site.uottawa.ca>

% 1) Minor editorial changes, as well as moving the information provided by

% Zhihao Chen to the new sections 3.1 and 4.1

%

% 2005/03/28 Zhihao Chen, CSE, USC, USA, <zhihaoch@cse.usc.edu>

% 1) Fix a mistake in line corresponding to cplx entry in the table of "The numeric values of the effort multipliers"

% "cplx 0.70 0.85 1.00 1.15 1.30 1.65 -1.86" should be

% "cplx 0.70 0.85 1.00 1.15 1.30 1.65 -2.36"

%

% 2) Additional information about various classifications of the projects are provided.

%

% 3) Additional usage information is provided

%

@data

Nominal,High,Very\_High,Nominal,Nominal,Low,Nominal,High,Nominal,Very\_High,Low,Nominal,High,Nominal,Low,70,278 % instance number: 1

Very\_High,High,High,Very\_High,Very\_High,Nominal,Nominal,Very\_High,Very\_High,Very\_High,Nominal,High,High,High,Low,227,1181 % instance number: 2

Nominal,High,High,Very\_High,High,Low,High,High,Nominal,High,Low,High,High,Nominal,Low,177.9,1248 % instance number: 3

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,115.8,480 % instance number: 4

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,29.5,120 % instance number: 5

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,19.7,60 % instance number: 6

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,66.6,300 % instance number: 7

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,5.5,18 % instance number: 8

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,10.4,50 % instance number: 9

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,14,60 % instance number: 10

Nominal,Nominal,High,High,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,16,114 % instance number: 11

High,Nominal,High,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,6.5,42 % instance number: 12

Nominal,Nominal,High,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,13,60 % instance number: 13

Nominal,Nominal,High,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,8,42 % instance number: 14

Nominal,Nominal,High,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,High,Nominal,High,High,High,Nominal,90,450 % instance number: 15

High,Nominal,Nominal,High,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Nominal,Nominal,Nominal,Nominal,15,90 % instance number: 16

High,Nominal,High,Nominal,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Nominal,Nominal,Nominal,Nominal,38,210 % instance number: 17

Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Nominal,Nominal,Nominal,Nominal,10,48 % instance number: 18

Nominal,Very\_High,High,Very\_High,Very\_High,Low,High,Very\_High,High,Nominal,Low,High,Very\_High,Very\_High,Low,161.1,815 % instance number: 19

Nominal,Very\_High,High,Very\_High,Very\_High,Low,High,Very\_High,High,Nominal,Low,High,Very\_High,Very\_High,Low,48.5,239 % instance number: 20

Nominal,Very\_High,High,Very\_High,Very\_High,Low,High,Very\_High,High,Nominal,Low,High,Very\_High,Very\_High,Low,32.6,170 % instance number: 21

Nominal,Very\_High,High,Very\_High,Very\_High,Low,High,Very\_High,High,Nominal,Low,High,Very\_High,Very\_High,Low,12.8,62 % instance number: 22

Nominal,Very\_High,High,Very\_High,Very\_High,Low,High,Very\_High,High,Nominal,Low,High,Very\_High,Very\_High,Low,15.4,70 % instance number: 23

Nominal,Very\_High,High,Very\_High,Very\_High,Low,High,Very\_High,High,Nominal,Low,High,Very\_High,Very\_High,Low,16.3,82 % instance number: 24

Nominal,Very\_High,High,Very\_High,Very\_High,Low,High,Very\_High,High,Nominal,Low,High,Very\_High,Very\_High,Low,35.5,192 % instance number: 25

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,25.9,117.6 % instance number: 26

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,24.6,117.6 % instance number: 27

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,7.7,31.2 % instance number: 28

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,9.7,25.2 % instance number: 29

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,2.2,8.4 % instance number: 30

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,3.5,10.8 % instance number: 31

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,8.2,36 % instance number: 32

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,Nominal,Nominal,High,High,Nominal,Low,66.6,352.8 % instance number: 33

Nominal,Low,High,Nominal,Extra\_High,Low,Low,High,Very\_High,Very\_High,Nominal,High,Nominal,Nominal,Nominal,150,324 % instance number: 34

Nominal,Low,High,Nominal,Nominal,Low,Low,High,Nominal,Nominal,Nominal,Very\_Low,Nominal,Nominal,Nominal,100,360 % instance number: 35

Nominal,Low,High,Nominal,Nominal,High,Low,High,High,High,Low,Very\_Low,Nominal,Nominal,Nominal,100,215 % instance number: 36

Nominal,Low,High,Nominal,Nominal,Low,Low,High,Very\_High,Very\_High,Nominal,High,Nominal,Nominal,Nominal,100,360 % instance number: 37

Nominal,Low,High,Nominal,Nominal,Low,Low,High,Very\_High,High,Nominal,High,Nominal,Nominal,Nominal,15,48 % instance number: 38

Nominal,Low,High,Nominal,Extra\_High,Low,Low,High,High,Nominal,Nominal,High,Nominal,Nominal,Nominal,32.5,60 % instance number: 39

Nominal,Low,High,Nominal,Nominal,Low,Low,High,High,High,Nominal,High,Nominal,Nominal,Nominal,31.5,60 % instance number: 40

Nominal,Low,High,Nominal,Nominal,Low,Low,High,Very\_High,High,Nominal,High,Nominal,Nominal,Nominal,6,24 % instance number: 41

Nominal,Low,High,Nominal,Nominal,Low,Low,High,Very\_High,Nominal,Nominal,Low,Nominal,Nominal,Nominal,11.3,36 % instance number: 42

Nominal,Low,High,Nominal,Nominal,Low,Low,High,Very\_High,Very\_High,Nominal,High,Nominal,Nominal,Nominal,20,72 % instance number: 43

Nominal,Low,High,Nominal,Nominal,Low,Low,High,Very\_High,High,Nominal,High,Nominal,Nominal,Nominal,20,48 % instance number: 44

High,Low,High,Extra\_High,Extra\_High,Low,High,High,High,High,Nominal,High,High,High,Nominal,7.5,72 % instance number: 45

High,Low,High,Nominal,Nominal,Low,Low,Nominal,Nominal,High,Nominal,Nominal,High,Very\_Low,Nominal,302,2400 % instance number: 46

High,Nominal,High,High,High,Low,High,Nominal,High,Nominal,Nominal,Nominal,Low,Very\_High,Nominal,370,3240 % instance number: 47

High,Nominal,High,High,High,Low,High,Nominal,High,Nominal,Nominal,Nominal,Low,Very\_High,Nominal,219,2120 % instance number: 48

High,Nominal,High,High,High,Low,High,Nominal,High,Nominal,Nominal,Nominal,Low,Very\_High,Nominal,50,370 % instance number: 49

High,Nominal,Very\_High,High,High,Low,High,High,Nominal,Nominal,High,High,Low,Very\_High,High,101,750 % instance number: 50

Nominal,Nominal,Nominal,Nominal,Nominal,Low,Nominal,High,Very\_High,Very\_High,Low,High,High,Nominal,Nominal,190,420 % instance number: 51

Nominal,Nominal,High,Nominal,High,Nominal,Nominal,High,High,Nominal,Nominal,High,High,Nominal,High,47.5,252 % instance number: 52

Very\_High,Nominal,Extra\_High,High,High,Low,Low,Nominal,High,Nominal,Nominal,Nominal,Low,High,Nominal,21,107 % instance number: 53

Low,Nominal,Nominal,Nominal,Nominal,Low,Low,High,High,Very\_High,Nominal,High,Low,Low,High,423,2300 % instance number: 54

High,High,Nominal,Nominal,Nominal,Low,Low,Nominal,High,High,Nominal,High,Nominal,Nominal,Nominal,79,400 % instance number: 55

High,High,Low,Nominal,Nominal,Nominal,High,High,High,Nominal,Nominal,Nominal,High,Nominal,Nominal,284.7,973 % instance number: 56

Nominal,High,Low,Nominal,Nominal,High,Nominal,High,High,Nominal,Nominal,Nominal,High,High,Nominal,282.1,1368 % instance number: 57

Nominal,High,High,Very\_High,Nominal,Nominal,High,High,High,High,Nominal,High,Low,Low,High,78,571.4 % instance number: 58

Nominal,High,High,Very\_High,Nominal,Nominal,High,High,High,High,Nominal,High,Low,Low,High,11.4,98.8 % instance number: 59

Nominal,High,High,Very\_High,Nominal,Nominal,High,High,High,High,Nominal,High,Low,Low,High,19.3,155 % instance number: 60