Contents

1 Algorithm

```
PROMISING= 1
USEFUL= 1.05
GRANULARITY=3
NOW= a description of current situation
CHANGES= a description of acceptable changes
NCHANGES=1

categorise all CHANGES attribute ranges into GRANULAIRTY percentiles

for all examples do
    if example in NOW+CHANGES
    then descritize all CHANGES attributes in example
             if example in RELEVANT
             then example.repeats++
             else example.repeats=1
              RELEVANT=RELEVANT+example
             fi
        fi
done

ALL= total number of outputs in RELEVANT

for each attribute/range (a/r) in CHANGES do
    for each class
    do
        BESTF= number of time a/r appears in best class in RELEVANT
        DELTA= best.score - class.score
        F    = number of time a/r appears in this class in RELEVANT
        if (DELTA*F/BESTF > PROMISING)
             then CANDIDATES[a/r]++
        fi
    done
done

BASELINE=0 for X in RELEVANT do BASELINE=+ X.class.score  done;

for CANDIDATE=subset of CANDIDATES of size NCHANGES do
    SELECTED= all RELEVANT that satisfy CANDIDATE
    CHANGE=0
    for for X in SELECTED do CHANGE=+ X.class.score  done;
        IMPROVEMENT[CANDIDATE]=CHANGE/BASELINE
done

return all CANDIDATES with IMPROVEMENT[CANDIDATE] > USEFUL
```

2 Promising ranges

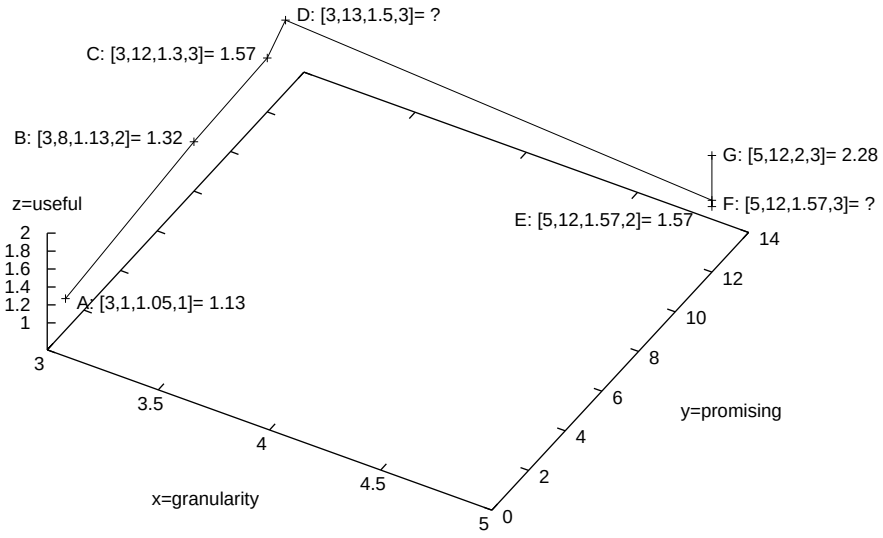
frequency counts of $CANDIDATES$ at $PROMISING = 1$

```
 1 :    1
 2 :    5 ~
 3 :   19 ~~~~~
 4 :   35 ~~~~~~~~~
 5 :   32 ~~~~~~~~
 6 :   51 ~~~~~~~~~~~~~~
 7 :   42 ~~~~~~~~~~~
 8 :   14 ~~~~
 9 :    6 ~~
10 :    1
11 :    3 ~
12 :    5 ~
13 :    1
17 :    1
```

3 Baseline

```
     score :    1
what if : no change?
   hilo :   36% ~~~~~~~~~
   lolo :   24% ~~~~~~
   hihi :   25% ~~~~~~
   lohi :   16% ~~~~
```

4 Flying with JANE



5 A

Notes: $NCHANGES > 1$ cancelled- too long

```
  score :    1
what if : no change?
   hilo :   36% ~~~~~~~~~
   lolo :   24% ~~~~~~
   hihi :   25% ~~~~~~
   lohi :   16% ~~~~


  score : 1.13526
what if : [[goodImplemenation@142]Cost=1]?
   hilo :   27% ~~~~~~~
   lolo :   31% ~~~~~~~~
   hihi :   20% ~~~~~
   lohi :   22% ~~~~~~

  score : 1.13278
what if : [[iv&v@76]Cost=1]?
   hilo :   23% ~~~~~~
   lolo :   39% ~~~~~~~~~~
   hihi :   15% ~~~~
```

```
   lohi :   24% ~~~~~~

[and some more]
```

6 B

$NCHANGES$ =2 ran ok

```
  score : 1.31552
what if : [[goodImplemenation@142]Cost=1, [goodProject@150]Cost=1]?
   hilo :   19% ~~~~~
   lolo :   34% ~~~~~~~~~
   hihi :   15% ~~~~
   lohi :   32% ~~~~~~~~

  score : 1.31051
what if : [[iv&v@76]Cost=1, [configurationManagement@73]Cost=1]?
   hilo :   16% ~~~~
   lolo :   44% ~~~~~~~~~~~
   hihi :    7% ~~
   lohi :   34% ~~~~~~~~~

 [and some more]
```

7 C

to run $NCHANGES = 3$, needed fewer $CANDIDATES$ so $PROMISING$ set to 12 LESSON: to increase $NCHANGES$, need tougher rules for $PROMISING$.

```
  score : 1.56854
what if : [[iv&v@76]Cost=1, [feasibleDesignToCode@127]Cost=1, [goodProject@150]Cost=1]?
   hilo :   11% ~~~
   lolo :   35% ~~~~~~~~~
   hihi :    8% ~~
   lohi :   46% ~~~~~~~~~~~~

  score : 1.49165
what if : [[goodImplemenation@142]Cost=1, [feasibleDesignToCode@127]Cost=1, [goodProject@
   hilo :    8% ~~
   lolo :   40% ~~~~~~~~~~
   hihi :   13% ~~~
   lohi :   40% ~~~~~~~~~~

  score : 1.45167
what if : [[iv&v@76]Cost=1, [goodImplemenation@142]Cost=1, [goodProject@150]Cost=1]?
   hilo :    7% ~~
   lolo :   48% ~~~~~~~~~~~~
   hihi :    5% ~
   lohi :   40% ~~~~~~~~~~

   [and some more]
```

8 D

Bad idea- can't increase promising over 13 and lookf ro $CANDIDATES$ of size 3- nothing there to see!

9 E

Increase granularity, more to search, so decrease $NCHANGES$.

Found a small change that looked exciting.

```
  score : 1.43308
what if : [[feasibleDesignToCode@127]Cost=1, [unitTest@134]Cost=1]?
   hilo :   12% ~~~
   lolo :   37% ~~~~~~~~~
   hihi :   14% ~~~~
   lohi :   37% ~~~~~~~~~
```

score : 1.40111
what if : [[problemReport@70]Cost=2, [goodTesting@136]Cost=2]?
hilo : 24% ~~~~~~
lolo : 19% ~~~~~
hihi : 22% ~~~~~~
lohi : 35% ~~~~~~~~~

```
  score : 1.38089
what if : [[unitTest@134]Cost=1, [problemReport@70]Cost=2]?
   hilo :   24% ~~~~~~
   lolo :   24% ~~~~~~
   hihi :   18% ~~~~~
   lohi :   35% ~~~~~~~~~
```

10 F

Too much output! Prune it!

11 G

Yee hah! Read it and weep.

```
  score : 2.28231
what if : [[testPlans@131]Cost=1, [problemReport@70]Cost=2, [goodTesting@136]Cost=2]?
   hilo :    0%
   lolo :    0%
   hihi :   25% ~~~~~~
   lohi :   75% ~~~~~~~~~~~~~~~~~~~

  score : 2.15189
what if : [[unitTest@134]Cost=1, [problemReport@70]Cost=2, [goodTesting@136]Cost=2]?
   hilo :   20% ~~~~~
   lolo :    0%
```

```
   hihi :    0%
   lohi :   80% ~~~~~~~~~~~~~~~~~~~

  score : 2.11929
what if : [[goodProject@150]Cost=1, [systemIntegration@145]Cost=1, [goodTesting@136]Cost=
   hilo :    0%
   lolo :   25% ~~~~~~
   hihi :    0%
   lohi :   75% ~~~~~~~~~~~~~~~~~~
```