Basic bash

& optimizing your SOAP assembly using a simple direct search approach

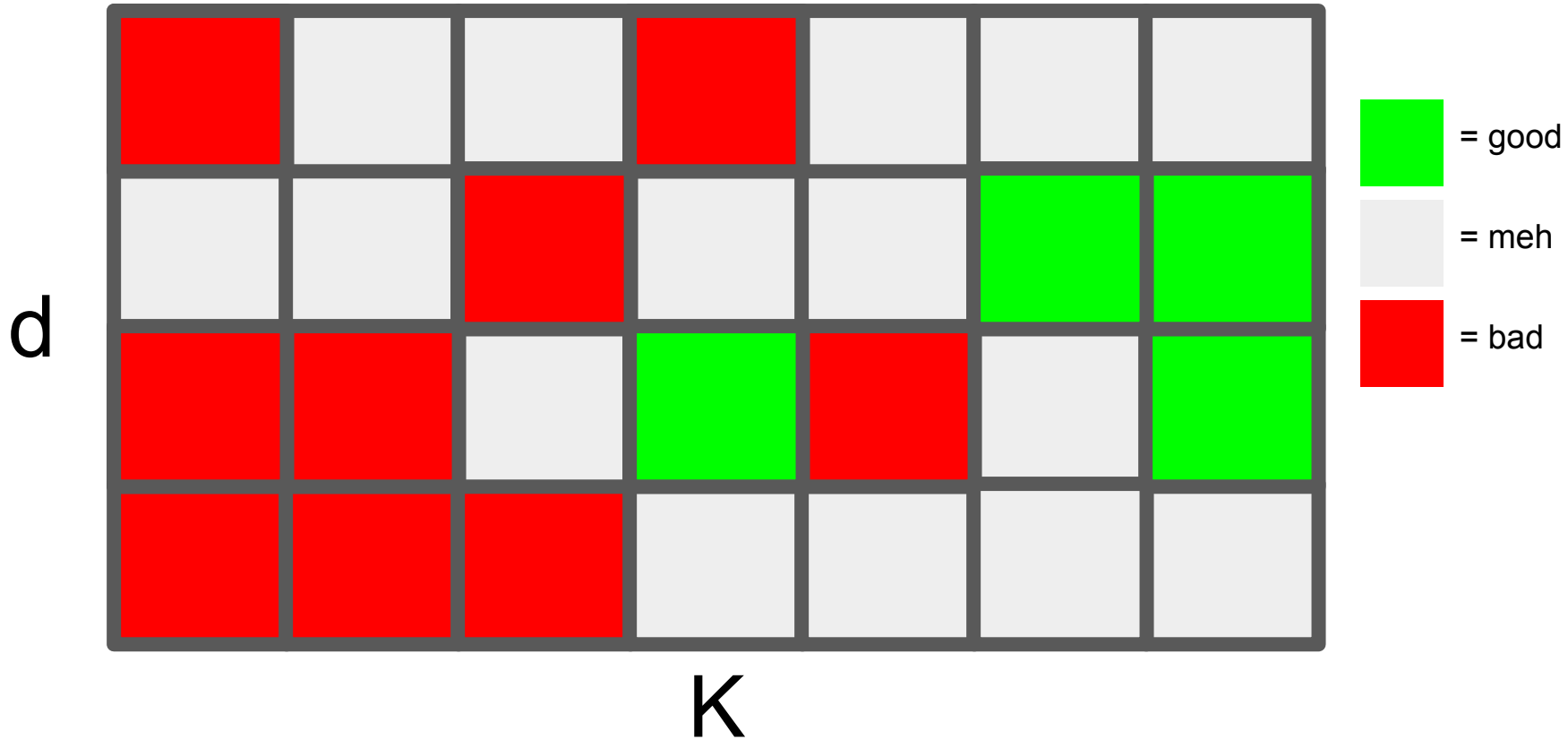
“The first rule of bash club: you don’t talk about bash club. The second rule: there are too many rules to relay to you here. You’ll see.”
-Tyler Durden (paraphrased)

The problem...

$ soapdenovo2-63mer all -s soap.config -K ?? -d ?? -o pablo.rules 1>err 2>log

- At least three parameters to select (really, at least one, more is ideal)
- Each could have quite dramatic effects on the quality of your assembly
- combinations of parameters could matter (one K does not rule them all)

Imagine a grid of soap parameter combinations...



My fingers hurt!

- “Good” assemblies are rare relative to “bad” assemblies.
- Even with only 4 -d values and 7 -K values, we have 28 possible soap runs to try -- what a buzzkill.
- What if we didn’t search “far” enough?


FIGHT
CLUB
BASH

We need to discuss a few things in bash

- store numeric variables and print their values
- for loops (everyone stay calm)
- if conditionals
- Modulo operator

You already know the rest!

store numeric variables and print their values

```
$ number=6 #save a variable
$ echo $number #print variable’s content
$ echo $number | wc #pass variable to commands
```

for loops (everyone, stay calm)

```
for(( i=0; i<=10; i++ ))
do
    echo $i
done
```

Tip: Start at the “core” and work outward. What’s the meat?

```
for(( i=0; i<=10; i++ ))
do
    for(( j=0; j<=3; j++ ))
     do
         echo $i $j
    done
done
```

Cool, now modify the previous script any way you'd like, and try to predict what change will result. Try to make a change that does not produce an error, but don't worry if it does.

if conditionals

if you like potato chips (true or false?)
then
        tell it to the world!
else
        get your head checked

```
likeChips="yes"

if [ $likeChips == "yes" ]
then
        echo "Hey everyone, I LOVE CHIPS!"
else
        echo "I don't feel well."
fi
```

Modulo operator

$ echo $((6 % 2))
$ 0
“2 goes into 6 a total of 3 times with a remainder of ***0***”

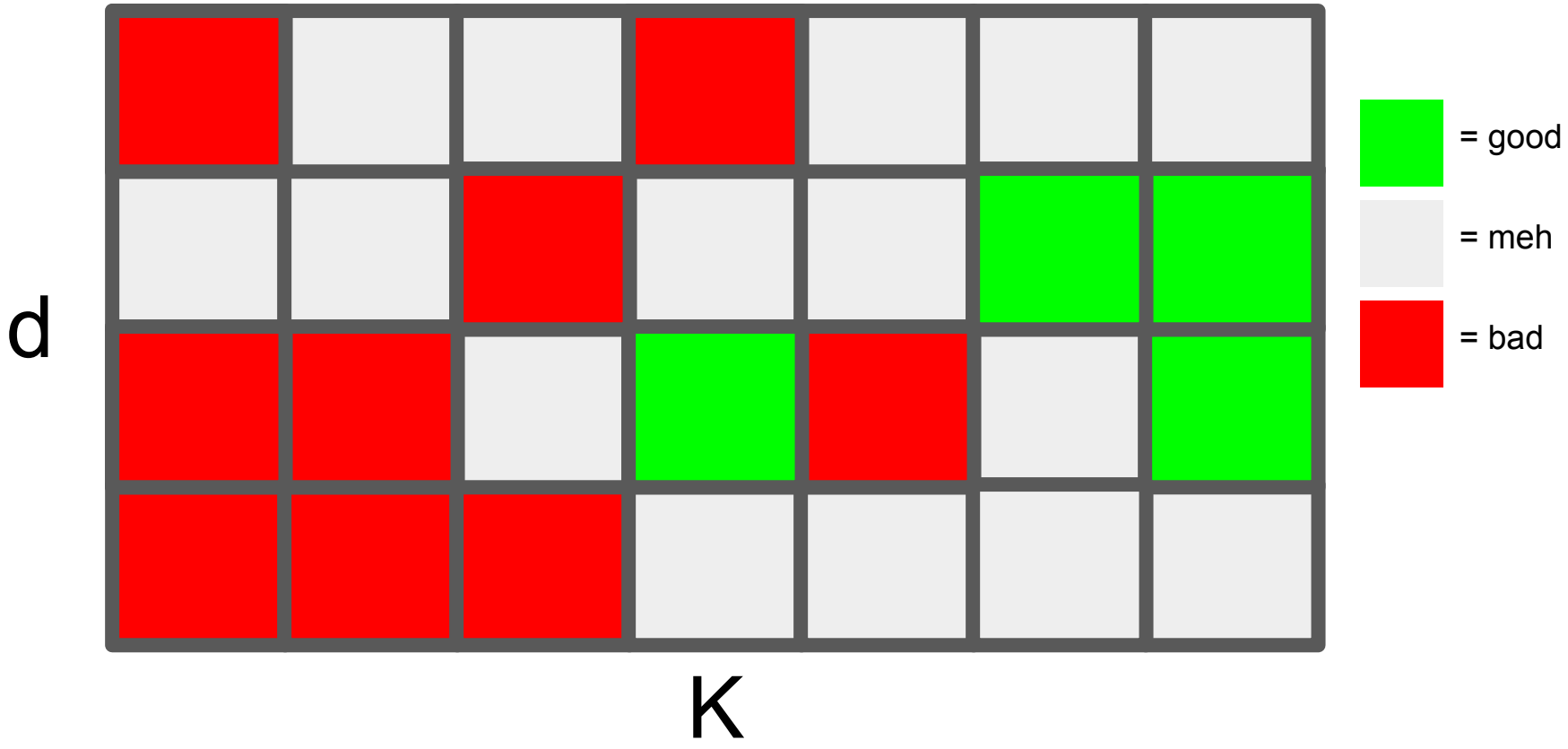
$ echo $((5 % 3))
$ 2
“3 goes into 5 a total of 1 time with a remainder of ***2***”

$ echo $((9 % 4))
$ 1
“4 goes into 9 a total of 2 times with a remainder of ***1***”

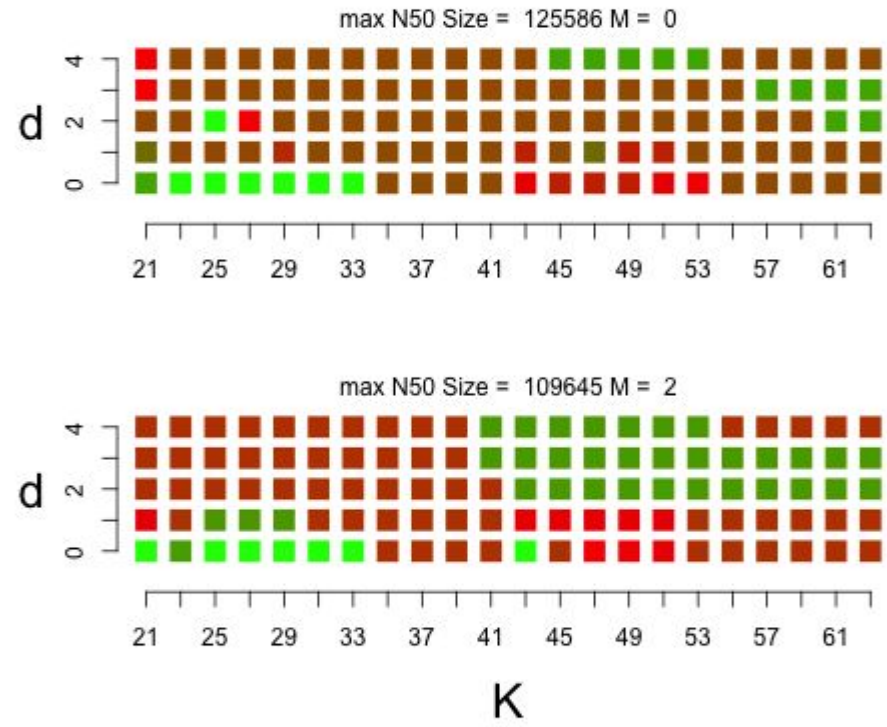
All together now!

```
for(( i=0; i<=10; i++ ))
do
    if [ $(( $i % 2 )) -ne 0 ] # -ne means not equal, like !=
    then
        for(( j=0; j<=3; j++ ))
        do
            echo $i $j
        done
    fi
done
```

We’re ready for soapGrid.sh!



$ git clone https://github.com/silastittes/soapSearch.git
$ cp soapSearch/soapGrid.sh ./
#check out the file, modify boundaries of parameters if need be.
$ bash soapGrid.sh <soap.config>


max N50 Size = 125586 M = 0
max N50 Size = 105336 M = 1
max N50 Size = 109645 M = 2
max N50 Size = 107958 M = 3
d
d
0
2
4
21
25
29
33
37
41
45
49
53
57
61
K
K