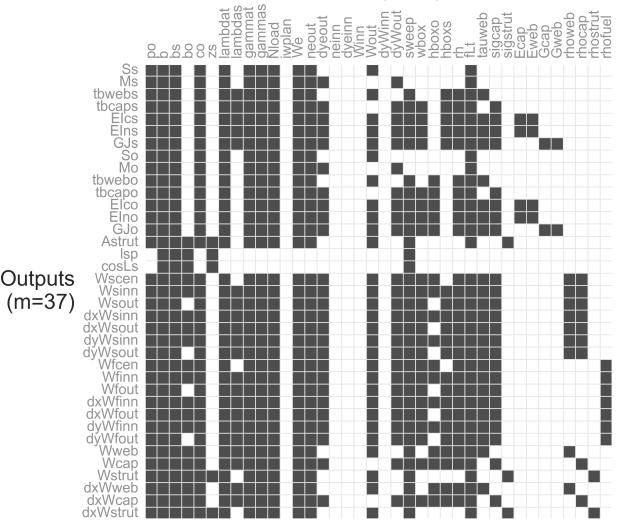


input_vector = [114115.099, 37.533, 10.697, ..., 2700.0, 817.0]

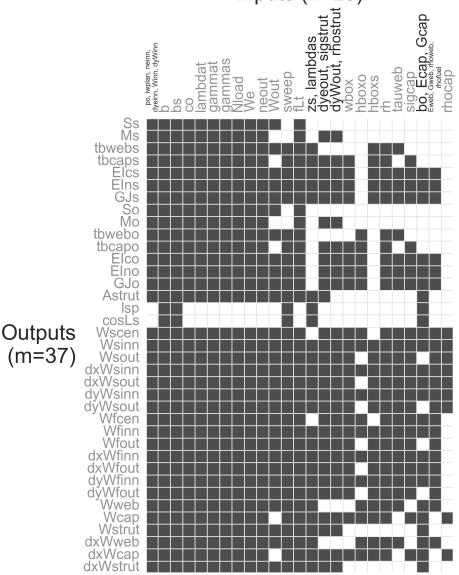
```
contaminated input vectors = [
          NaN, 37.533, 10.697, ..., 2700.0, 817.0],
   [114115.099, NaN, 10.697, ..., 2700.0, 817.0],
  [114115.099, 37.533, NaN, ..., 2700.0, 817.0],
  [114115.099, 37.533, 10.697, ..., NaN, 817.0],
  [114115.099, 37.533, 10.697, \ldots, 2700.0, NaN],
```

contaminated_outputs[18] = [*NaN*, 3475012.2, *NaN*, ..., 170545.5, *NaN*]

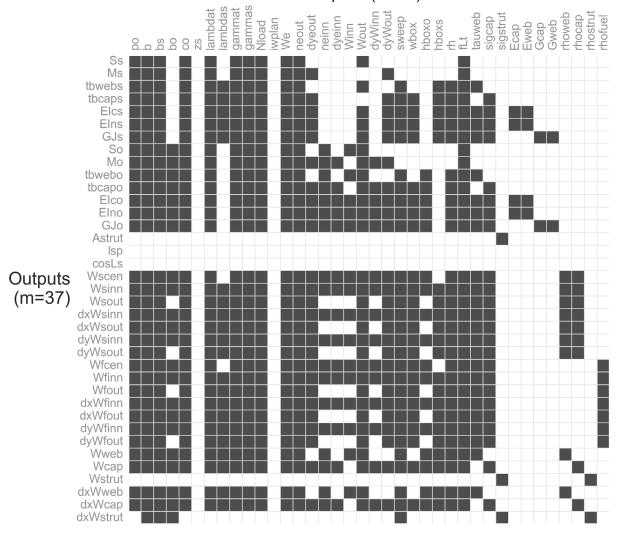
Sparsity pattern of `surfw()`, using NaN-contamination Inputs (n=38)



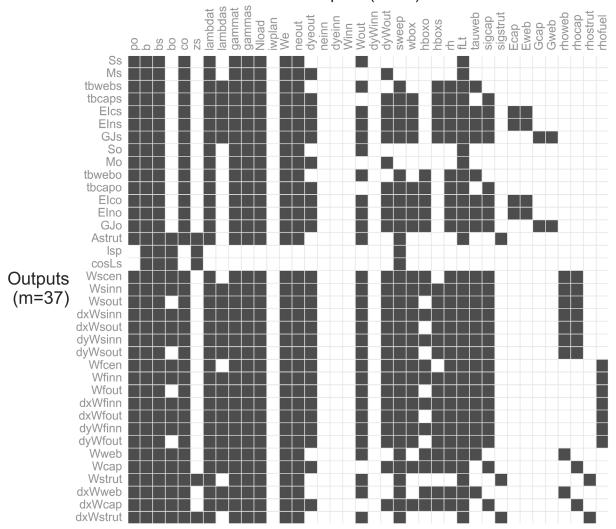
Column-compressed sparsity pattern of `surfw()` Inputs (n=25)



Sparsity pattern of `surfw()`, using NaN-contamination Inputs (n=38)

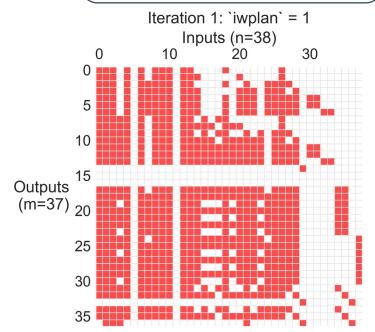


Sparsity pattern of `surfw()`, using NaN-contamination Inputs (n=38)



Step 1:

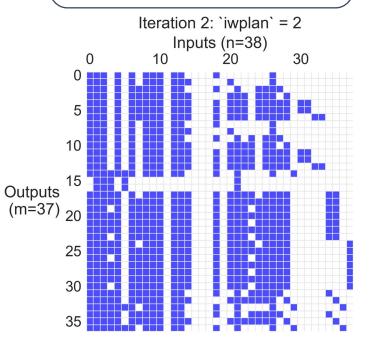
- Trace sparsity with NaN-propagation
- Compute gradient, take optimization step, etc.





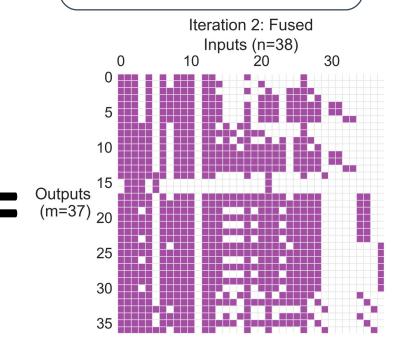
Step 2:

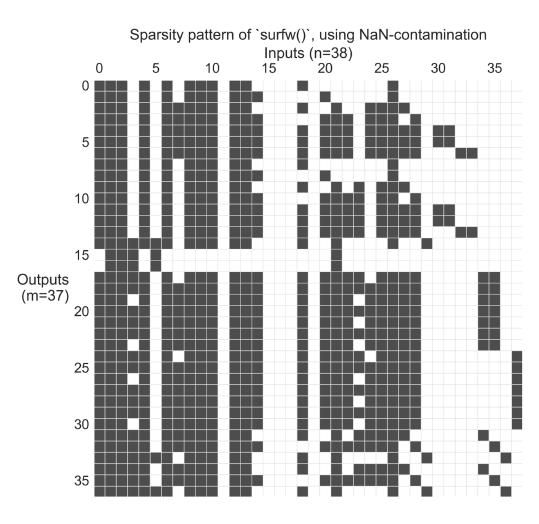
 At next iteration, if any new values for discrete variables are seen, redo the sparsity trace



Step 3:

 Take the union of the new sparsity and the previous one, and use that going forward





X = false negatives!

