NONLINEAR MODELING IN R WITH GAMS

2-Dimensional Smooths and Spatial Data

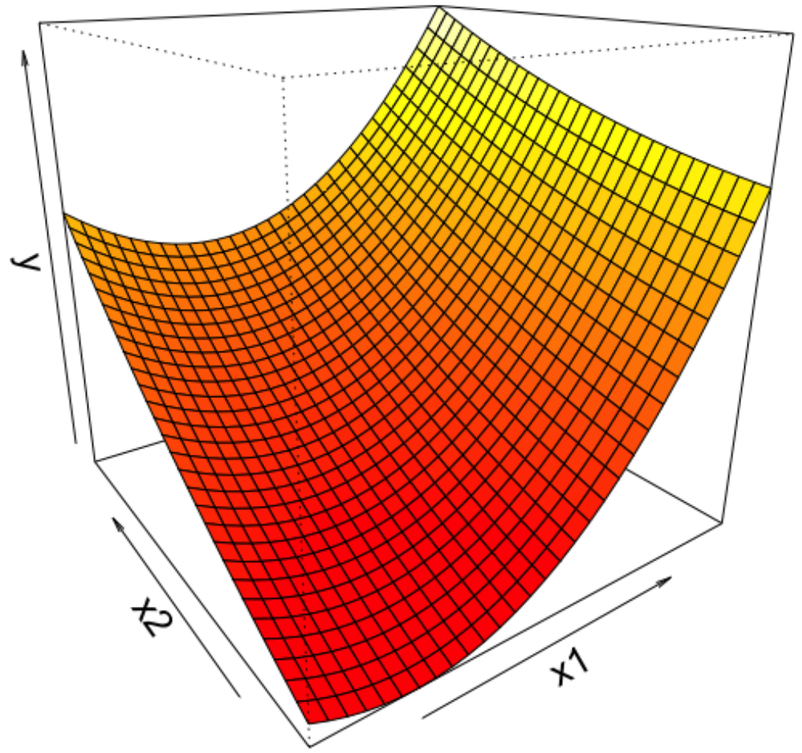
Noam Ross
Senior Research Scientist, EcoHealth Alliance

Interactions

$$y = \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$$

Interactions in GAMs

$$y = s(x_1, x_2)$$



Syntax for interactions

```
gam(y ~ s(x1, x2), # <-- 2 variables
        data = dat, method = "REML")
```

Mixing interaction and single terms

```
gam(y ~ s(x1, x2) + s(x3),
        data = dat, method = "REML")
```

```
gam(y ~ s(x1, x2) + x3 + x4,
        data = dat, method = "REML")
```

Interaction model outputs

```
Family: gaussian
Link function: identity

Formula:
y ~ s(x1, x2)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.34256    0.01646   20.82   <2e-16 ***
 ---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
           edf Ref.df     F p-value
s(x1,x2) 10.82   14.9 14.37  <2e-16 *** #<-- Interaction
 ---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.519   Deviance explained = 54.5%
GCV = 0.057564  Scale est. = 0.054161  n = 200
```

Spatial data

```
meuse
```

```
          x      y cadmium copper lead zinc   elev       dist   om ffreq soil lime landuse dist.m
1    181072 333611    11.7     85  299 1022  7.909 0.00135803 13.6     1    1    1      Ah     50
2    181025 333558     8.6     81  277 1141  6.983 0.01222430 14.0     1    1    1      Ah     30
3    181165 333537     6.5     68  199  640  7.800 0.10302900 13.0     1    1    1      Ah    150
4    181298 333484     2.6     81  116  257  7.655 0.19009400  8.0     1    2    0      Ga    270
5    181307 333330     2.8     48  117  269  7.480 0.27709000  8.7     1    2    0      Ah    380
6    181390 333260     3.0     61  137  281  7.791 0.36406700  7.8     1    2    0      Ga    470
7    181165 333370     3.2     31  132  346  8.217 0.19009400  9.2     1    2    0      Ah    240
8    181027 333363     2.8     29  150  406  8.490 0.09215160  9.5     1    1    0      Ab    120
9    181060 333231     2.4     37  133  347  8.668 0.18461400 10.6     1    1    0      Ab    240
10   181232 333168     1.6     24   80  183  9.049 0.30970200  6.3     1    2    0       W    420
```

```
?sp::meuse
```

NONLINEAR MODELING IN R WITH GAMS

Let's practice!

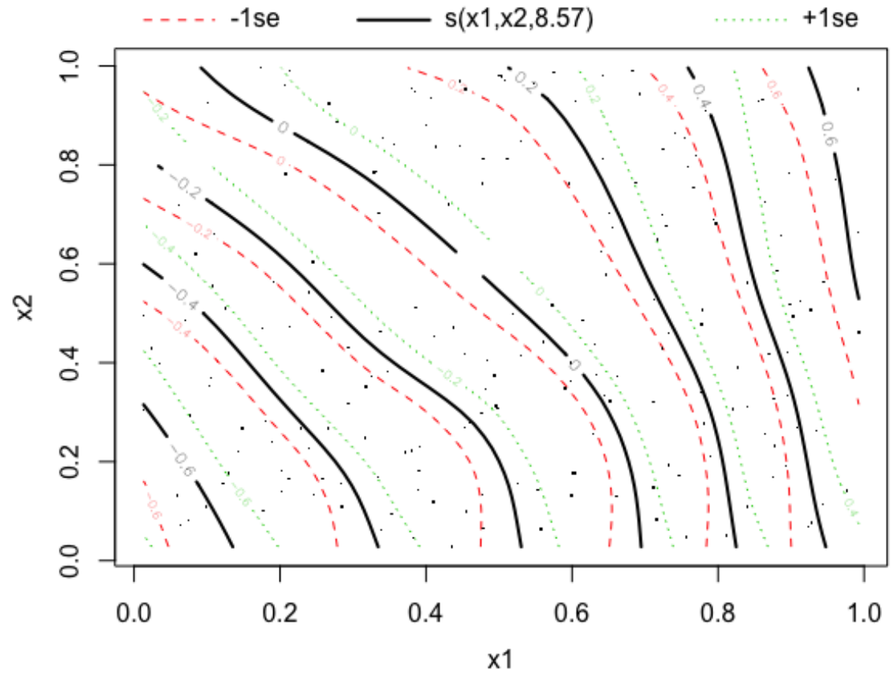
NONLINEAR MODELING IN R WITH GAMS

Plotting GAM interactions

Noam Ross
Senior Research Scientist, EcoHealth Alliance

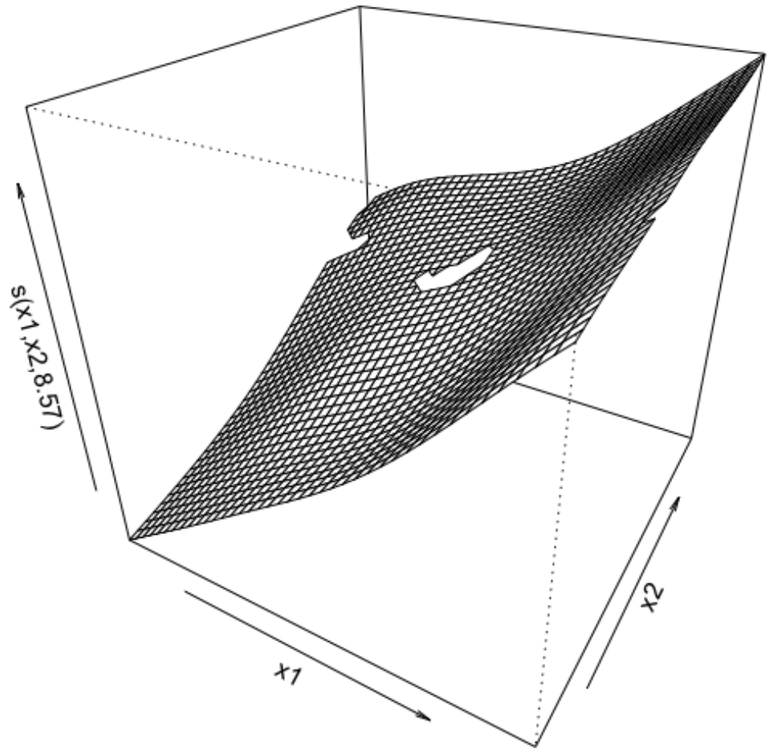
Using mgcv's plot() command with interactions.

```
plot(mod_2d)
```



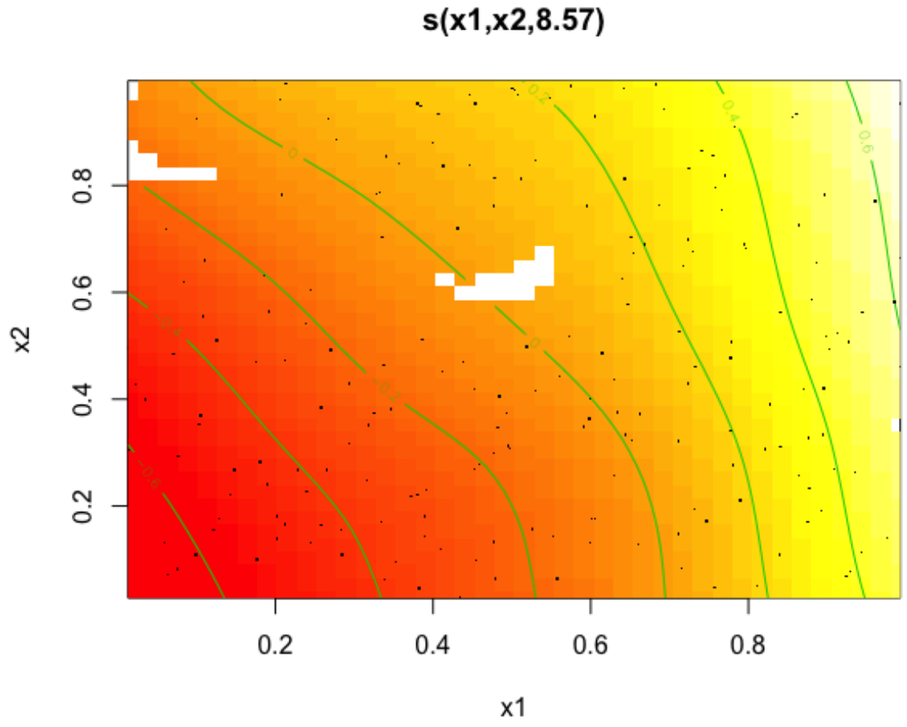
Using mgcv's plot() with interactions

```
plot(mod_2d, scheme = 1)
```



Using mgcv's plot() with interactions

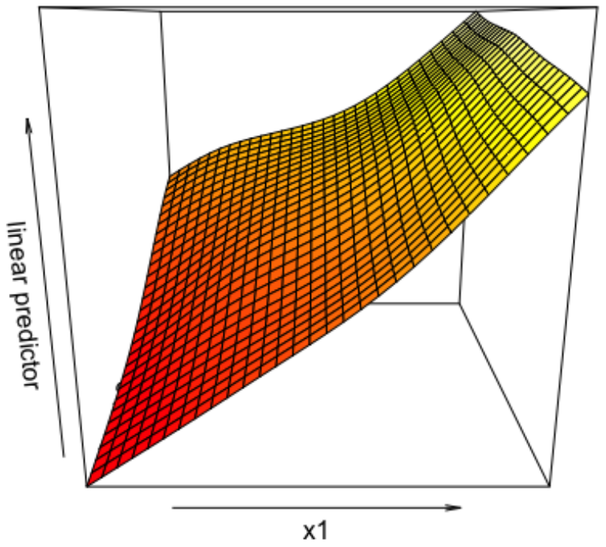
```
plot(mod_2d, scheme = 2)
```



Customizing interaction plots with vis.gam()

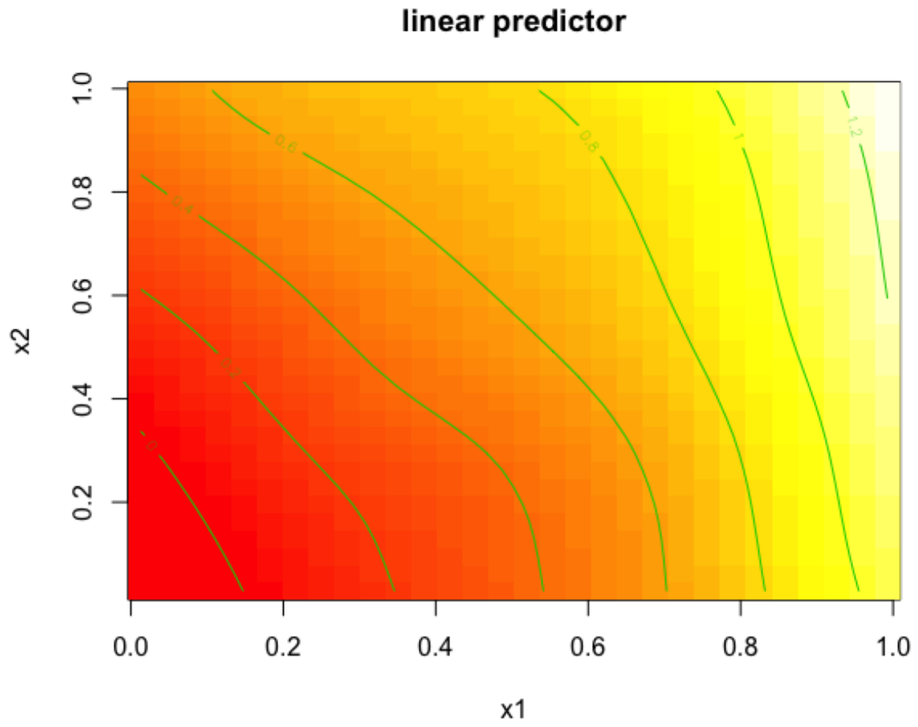
Customizing interaction plots with vis.gam()

```
vis.gam(x = mod,                    # GAM object
        view = c("x1", "x2"),       # variables
        plot.type = "persp")        # kind of plot
```



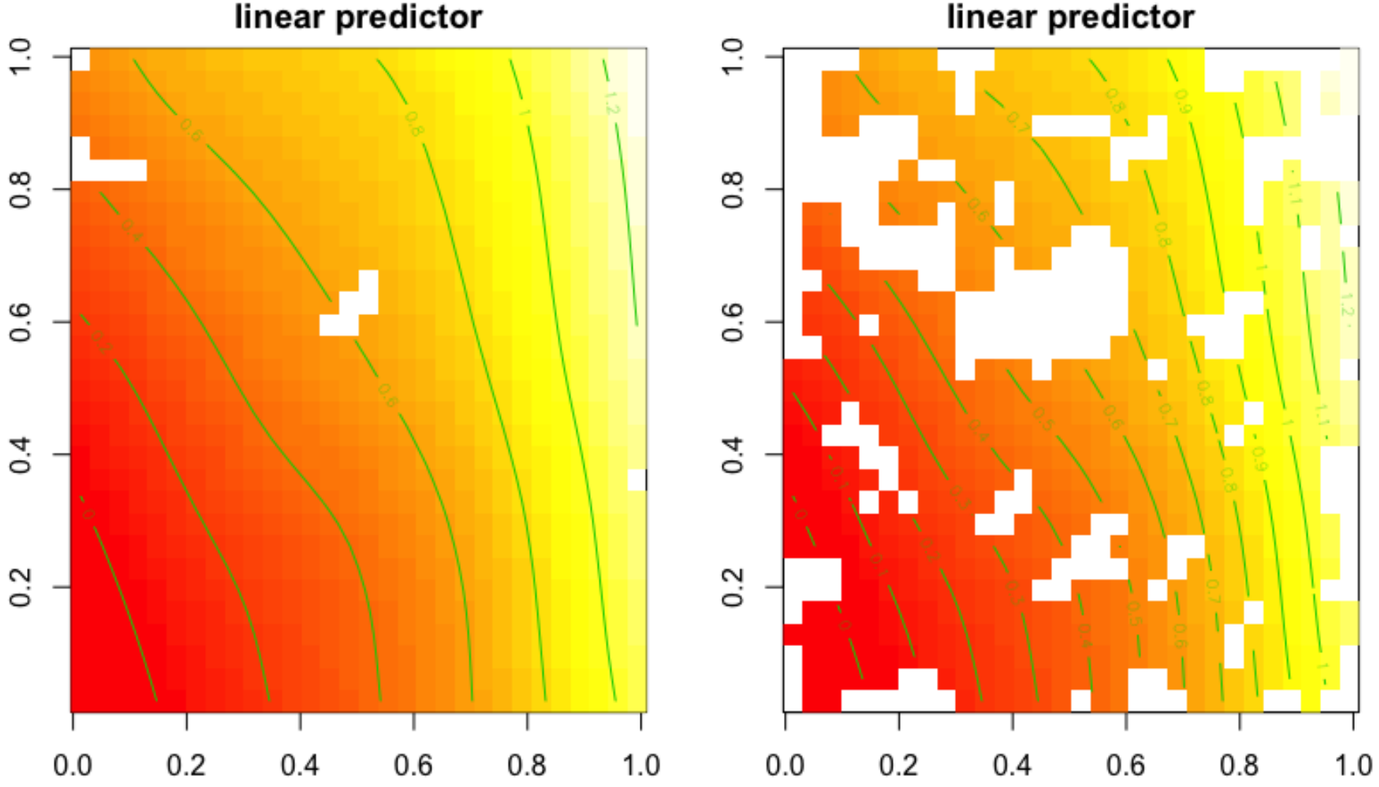
Customizing interaction plots with vis.gam() (2)

```
vis.gam(x = mod,                    # GAM object
        view = c("x1", "x2"),       # variables
        plot.type = "contour")      # kind of plot
```



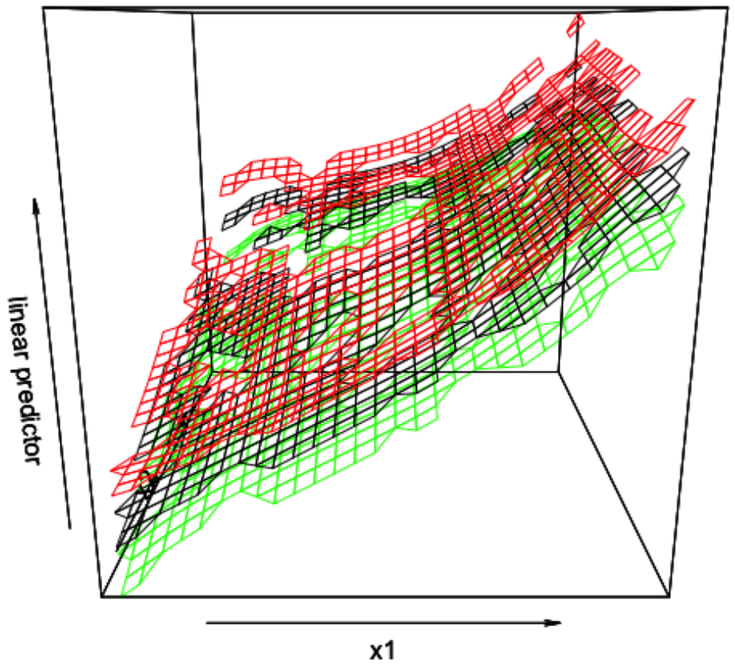
Customizing interaction plots with vis.gam()

```
vis.gam(mod, view = c("x1", "x2"), plot.type = "contour", too.far = 0.1)
vis.gam(mod, view = c("x1", "x2"), plot.type = "contour", too.far = 0.05)
```



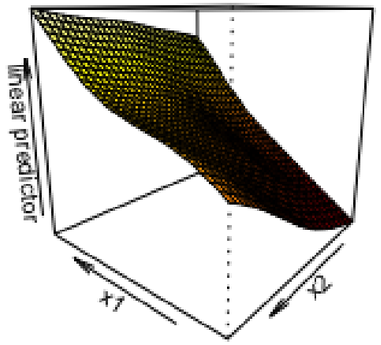
Options for perspective plots

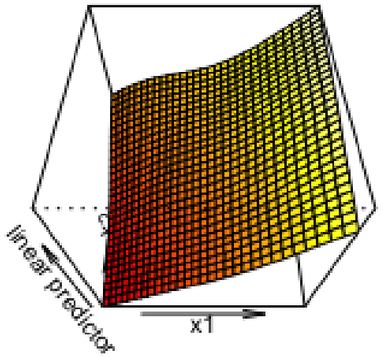
```
vis.gam(x = mod, view = c("x1", "x2"),
        plot.type = "persp", se = 2)
```

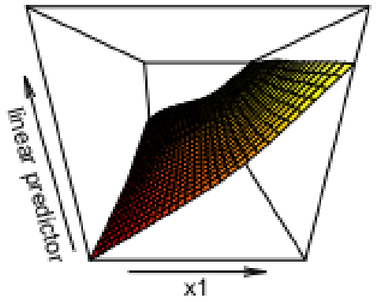


Options for perspective plots

```
vis.gam(g, view = c("x1", "x2"), plot.type = "persp", theta = 220)
vis.gam(g, view = c("x1", "x2"), plot.type = "persp", phi = 55)
vis.gam(g, view = c("x1", "x2"), plot.type = "persp", r = 0.1)
```

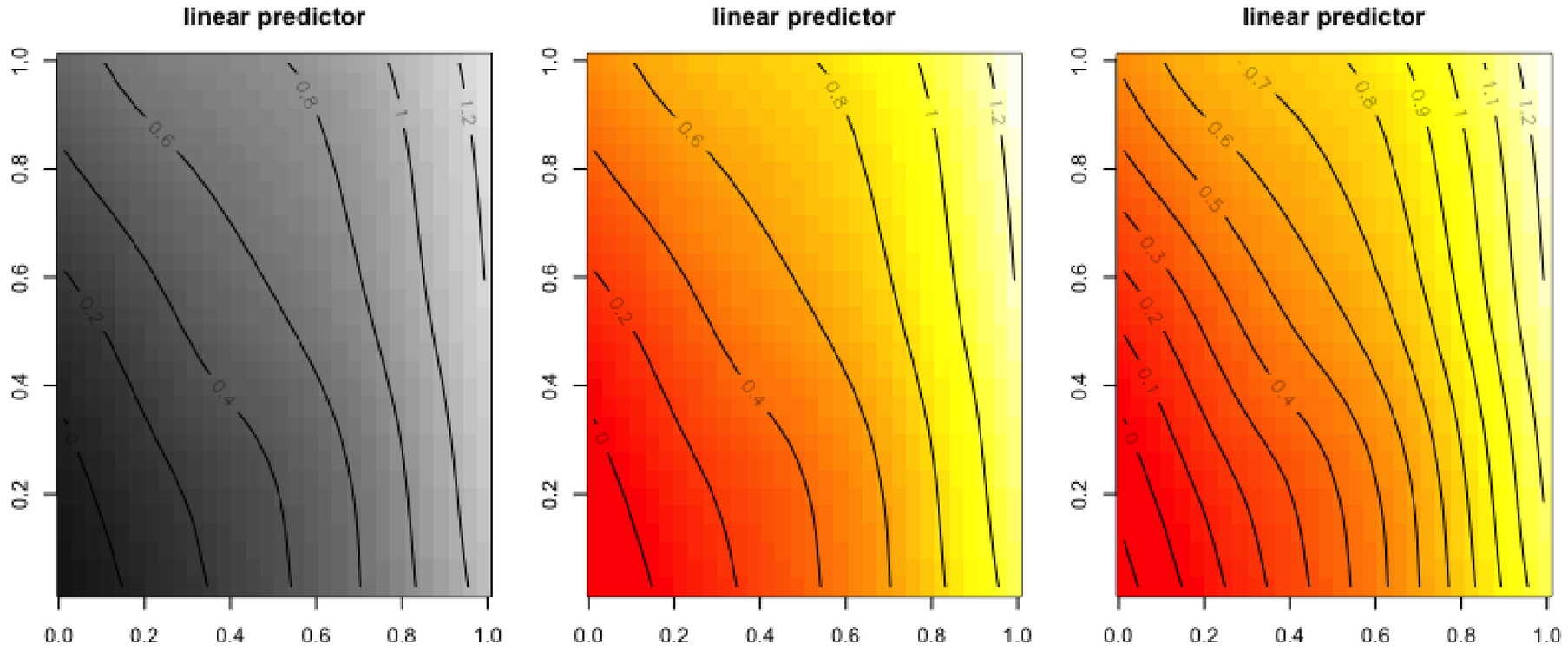






Options for contour plots

```
vis.gam(g, view = c("x1", "x2"), plot.type = "contour", color = "gray")
vis.gam(g, view = c("x1", "x2"), plot.type = "contour", contour.col = "blue")
vis.gam(g, view = c("x1", "x2"), plot.type = "contour", nlevels = 20)
```



NONLINEAR MODELING IN R WITH GAMS

Now let's make some plots!

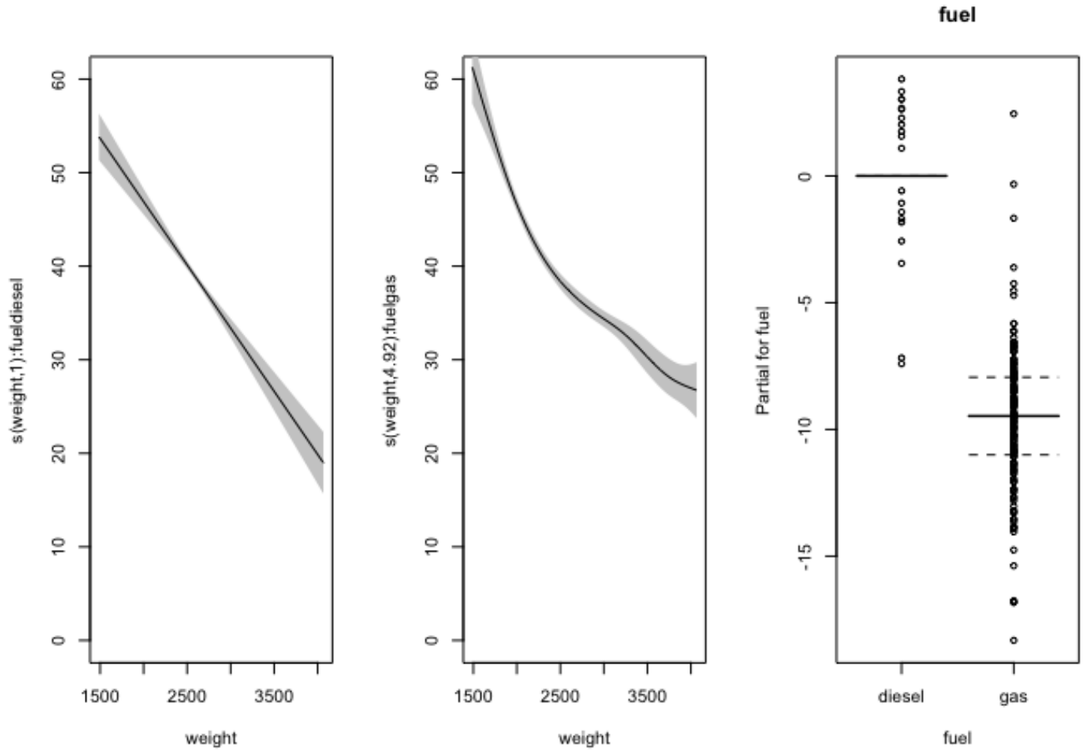
NONLINEAR MODELING IN R WITH GAMS

Visualizing categorical-continuous interactions

Noam Ross
Senior Research Scientist, EcoHealth Alliance

Categorical-continuous interactions

```
model4b <- gam(hw.mpg ~ s(weight, by = fuel) + fuel, data = mpg,
               method = "REML")
```



Factor-smooths

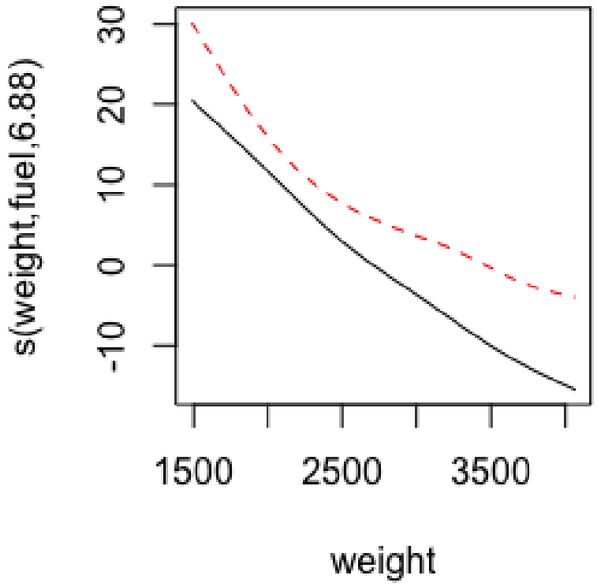
```
model4c <- gam(hw.mpg ~ s(weight, fuel, bs = "fs"),
               data = mpg,
               method = "REML")
```

Factor-smooths

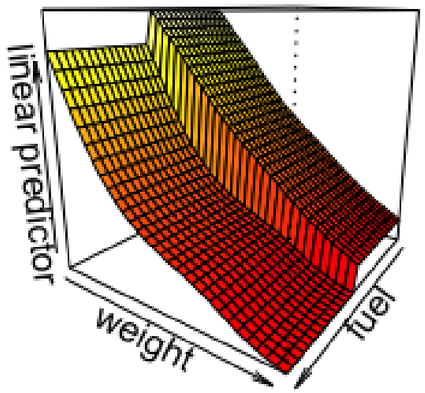
Plotting factor-smooths

```
plot(model4c)
vis.gam(model4c, theta = 125, plot.type = "persp")
```

Using plot()



Using vis.gam()



NONLINEAR MODELING IN R WITH GAMS

Let's practice!

NONLINEAR MODELING IN R WITH GAMS

Interactions with Different Scales

Noam Ross
Senior Research Scientist, EcoHealth Alliance

Interactions with one smoothing parameter

$$y = s(x_1, x_2)$$

$$\text{with smoothing parameter } \lambda$$

Variables with different scales or wiggliness

Numeric terms from `meuse` on different scales:

```
        x      y  elev    om
 1 181072 333611  7.91  13.6
 2 181025 333558  6.98  14
 3 181165 333537  7.8   13
 4 181298 333484  7.66  8
 5 181307 333330  7.48  8.7
 6 181390 333260  7.79  7.8
 7 181165 333370  8.22  9.2
 8 181027 333363  8.49  9.5
 9 181060 333231  8.67  10.6
10 181232 333168  9.05  6.3
```

Tensor Smooths

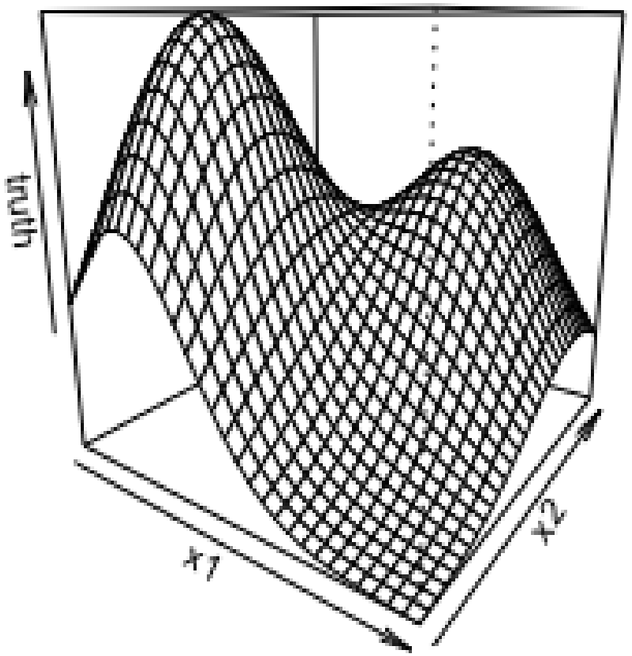
$$y = te(x_1, x_2)$$

$$\text{with smoothing parameters } \lambda_1, \lambda_2$$

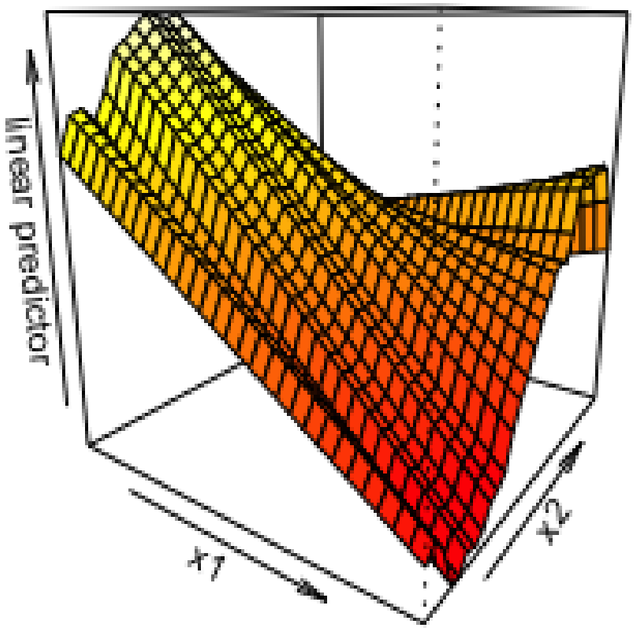
```
gam(y ~ te(x1, x2), data = data, method = "REML")
```

```
gam(y ~ te(x1, x2, k = c(10, 20)), data = data, method = "REML")
```

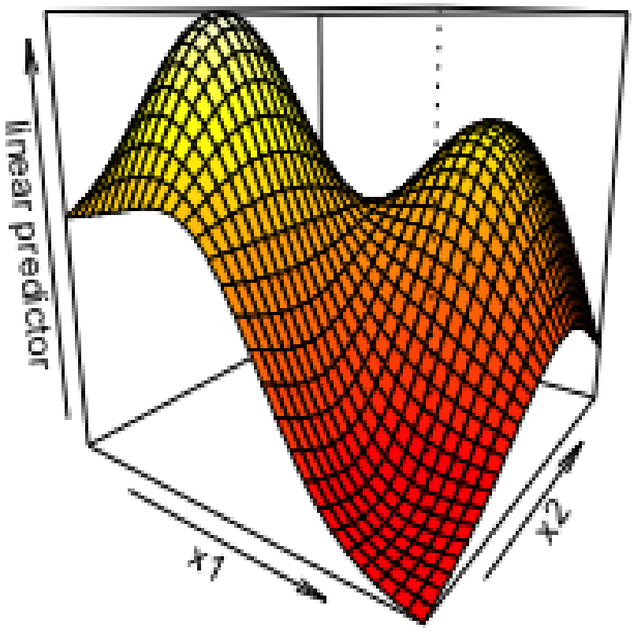
Actual Relationship



Fit using s()



Fit using te()



Tensor interactions

$$y = s(x_1) + s(x_2) + ti(x_1, x_2)$$

with smoothing parameters $\lambda_1, \lambda_2, \lambda_3, \lambda_4$

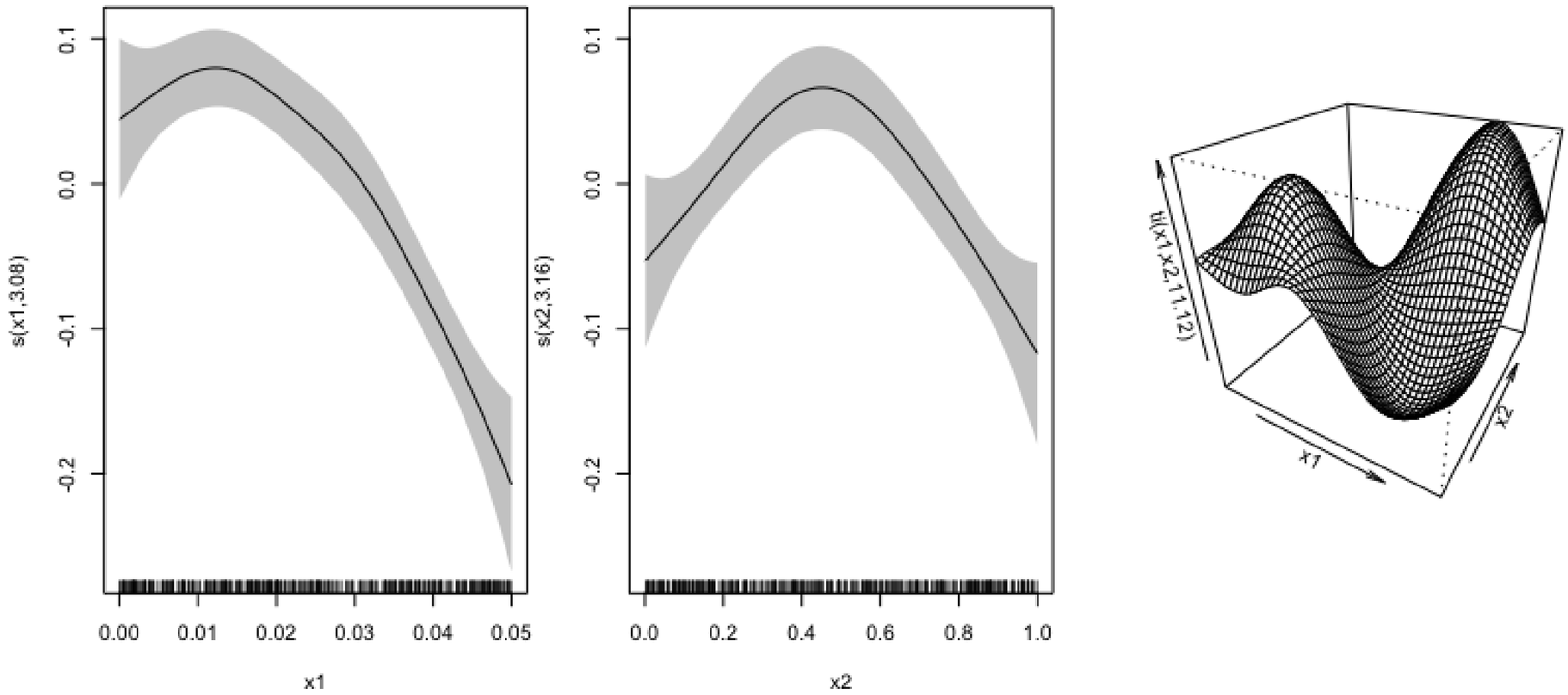
```
gam(y ~ s(x1) + s(x2) + ti(x1, x2), data = data, method = "REML")
```

Example: Tensor Interactions

```
Family: gaussian
Link function: identity

Formula:
y ~ s(x1) + s(x2) + ti(x1, x2)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.318698   0.008697   36.65   <2e-16 ***
 ---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
            edf Ref.df     F  p-value
te(x1)      4.93  6.009 23.16  < 2e-16 ***    # Separate terms for
te(x2)      3.42  4.242 10.35 2.75e-08 ***    # each variable and
ti(x1,x2) 10.15 12.763 16.08  < 2e-16 ***     # the interaction
 ---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.444   Deviance explained = 46.5%
-REML = -85.566  Scale est. = 0.037067  n = 500
```

Example: Tensor Interactions

```
gam(y ~ s(x1) + s(x2) + ti(x1, x2), data = data, method = "REML")
```



NONLINEAR MODELING IN R WITH GAMS

Let's practice!