flashlight: shedding light into black boxes

Alicja Gosiewska

27.01.2020, Warsaw

Presentation inspired by Using flashlight[1)] vignette prepared by Michael Mayer.

This is the purpose of the R package `flashlight`, which is inspired by the beautiful `DALEX` package.

1) https://github.com/mayer79/flashlight/blob/master/vignettes/flashlight.Rmd

Pros:

- Many model-agnostic tools

- Assessing multiple models in parallel without any redundancy in the code.

- Supporting "group by" operations.

- All methods are able to utilize case weights.

Currently, models with numeric or binary response are supported.

Flashlight

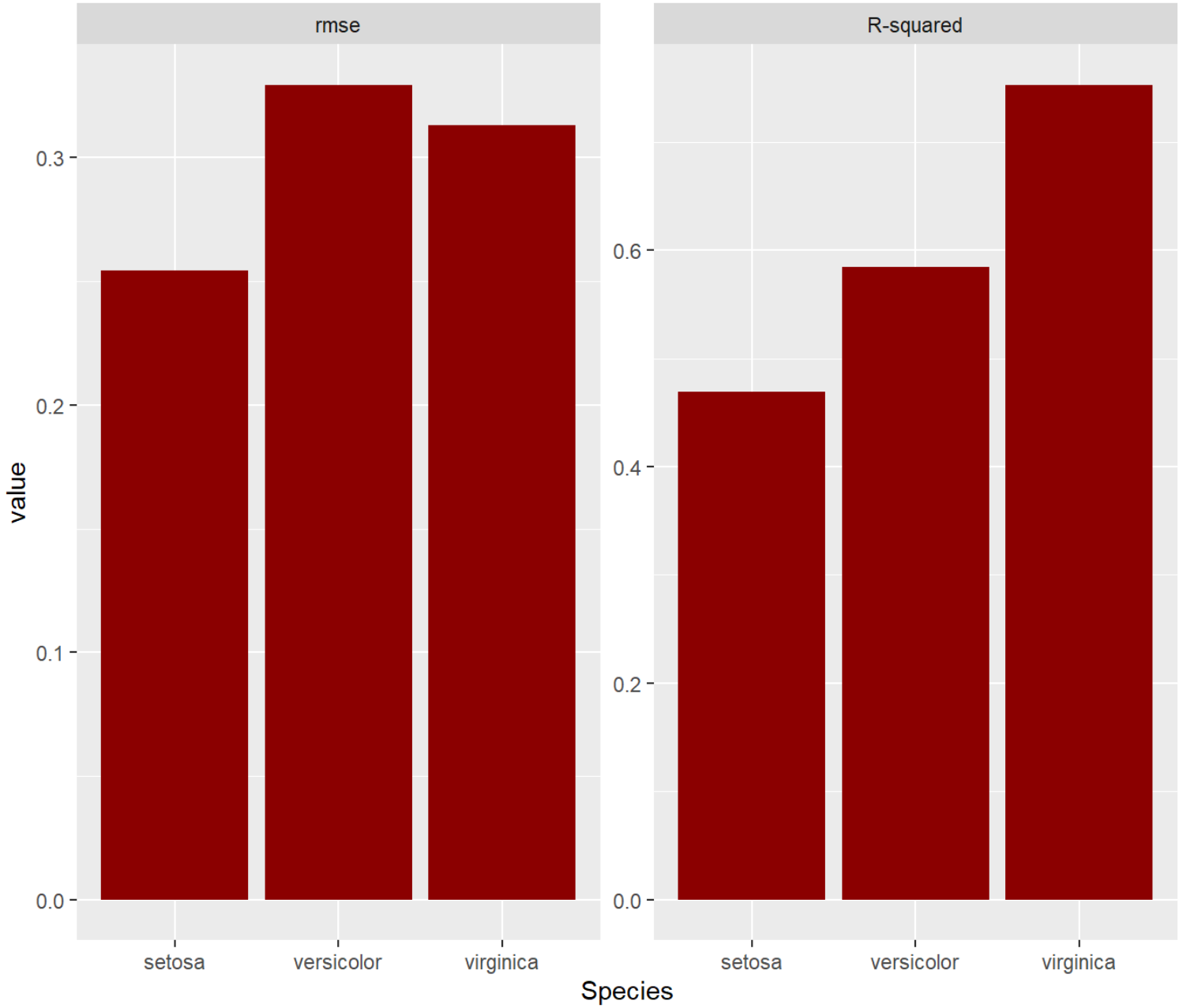
```r
# Fit model
fit <- lm(Sepal.Length ~ ., data = iris)

# Make flashlight
fl <- flashlight(model = fit, data = iris, y = "Sepal.Length", label = "ols",
                 metrics = list(rmse = rmse, `R-squared` = r_squared))

# Performance: rmse and R-squared
plot(light_performance(fl), fill = "darkred")
```

```r
fl_mean <- flashlight(model = mean(train$log_price), label = "mean",
                      predict_function = function(mod, X) rep(mod, nrow(X)))
fl_lm <- flashlight(model = fit_lm, label = "lm",
                    predict_function = function(mod, X) predict(mod, prep_lm(X)))
fl_rf <- flashlight(model = fit_rf, label = "rf",
                    predict_function = function(mod, X) predict(mod, X)$predictions)
fl_xgb <- flashlight(model = fit_xgb, label = "xgb",
                     predict_function = function(mod, X) predict(mod, prep_xgb(X, x)))
```

```r
fls <- multiflashlight(list(fl_mean, fl_lm, fl_rf, fl_xgb), y = "log_price", linkinv = exp,
                       data = valid, metrics = list(rmse = rmse, `R-squared` = r_squared))
```

Model performance

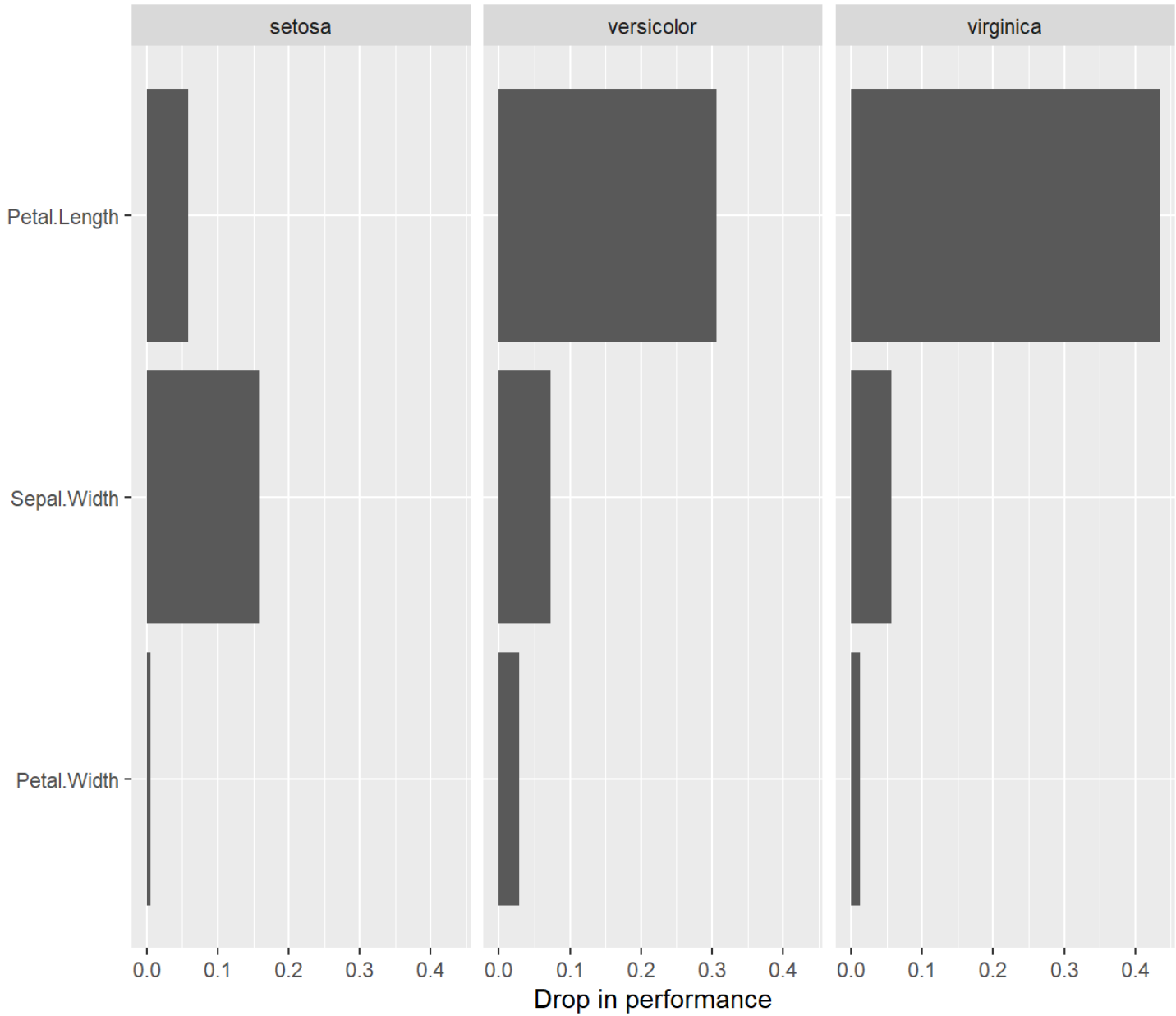
```
plot(light_performance(fl, by = "Species"), fill = "darkred")
```



Variable importance
by rmse

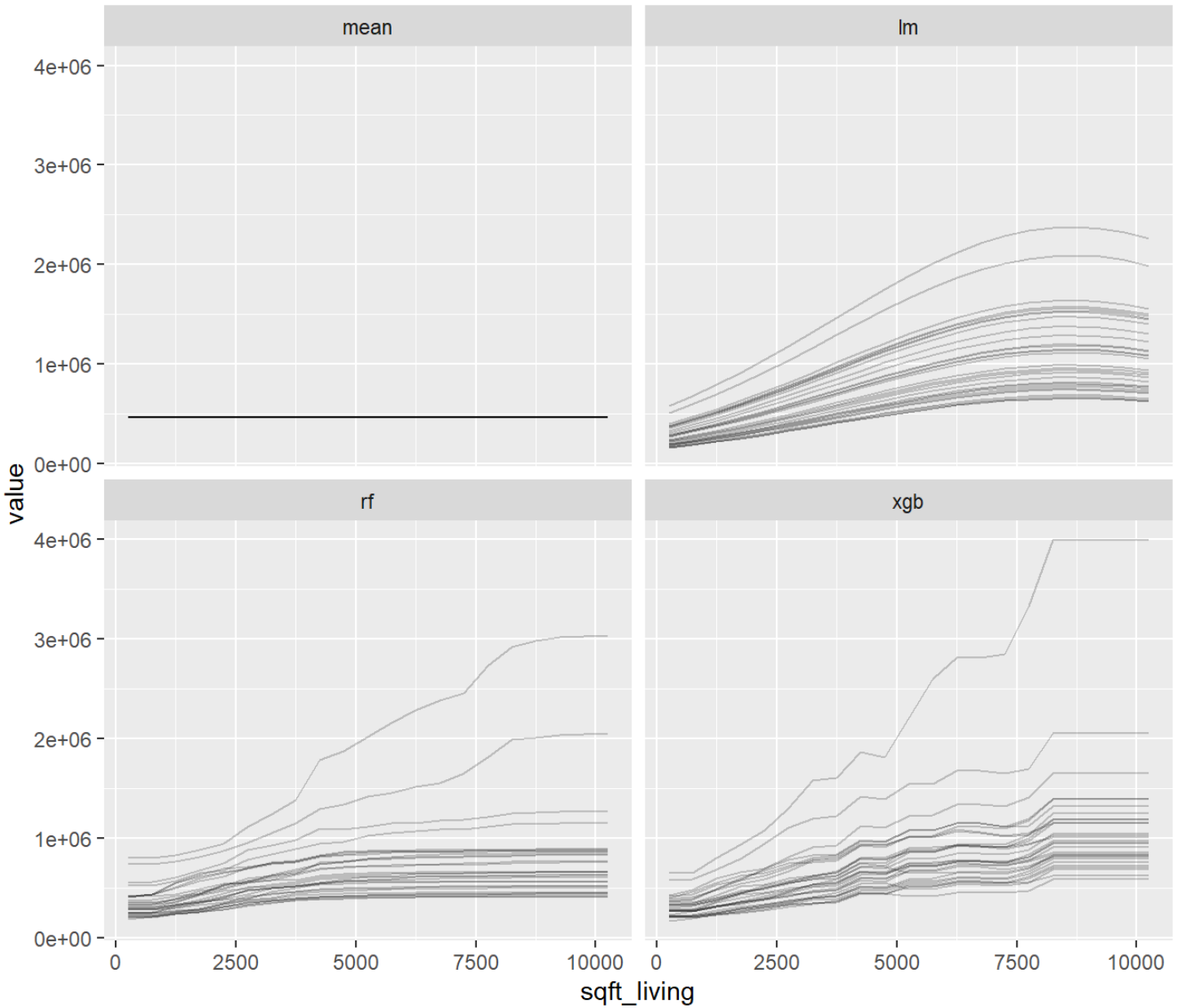
The algorithm measures importance of variable v as the drop in performance by permuting the values.

```
plot(light_importance(fl, by = "Species")) +
   scale_fill_viridis_d(begin = 0.2, end = 0.8)
```



Effects of input variables
ICE curves (ceteris paribus)

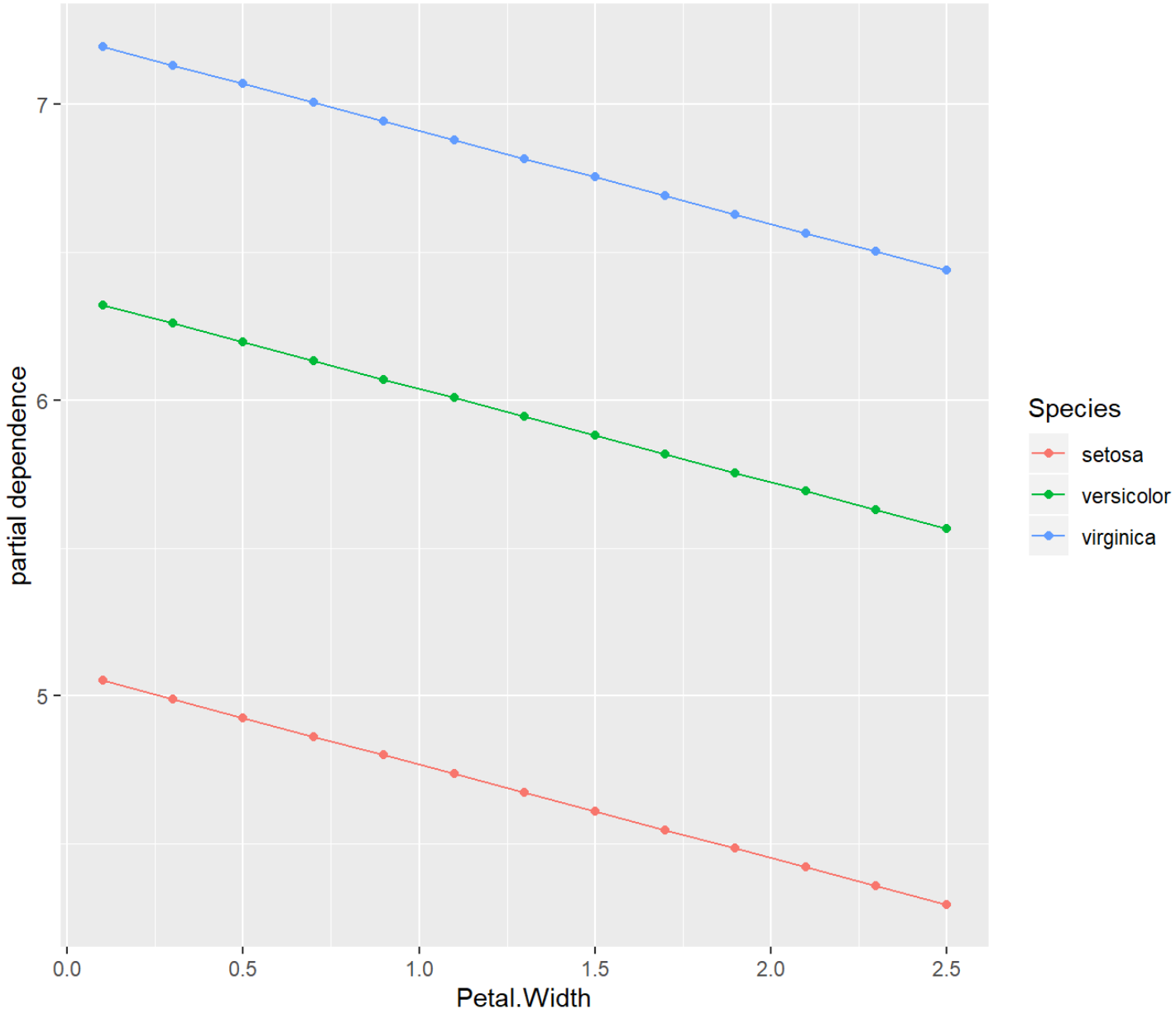
```
cp <- light_ice(fls, v = "sqft_living", n_max = 30, seed = 35)
plot(cp, alpha = 0.2)
```



Effects of input variables

PDP plot

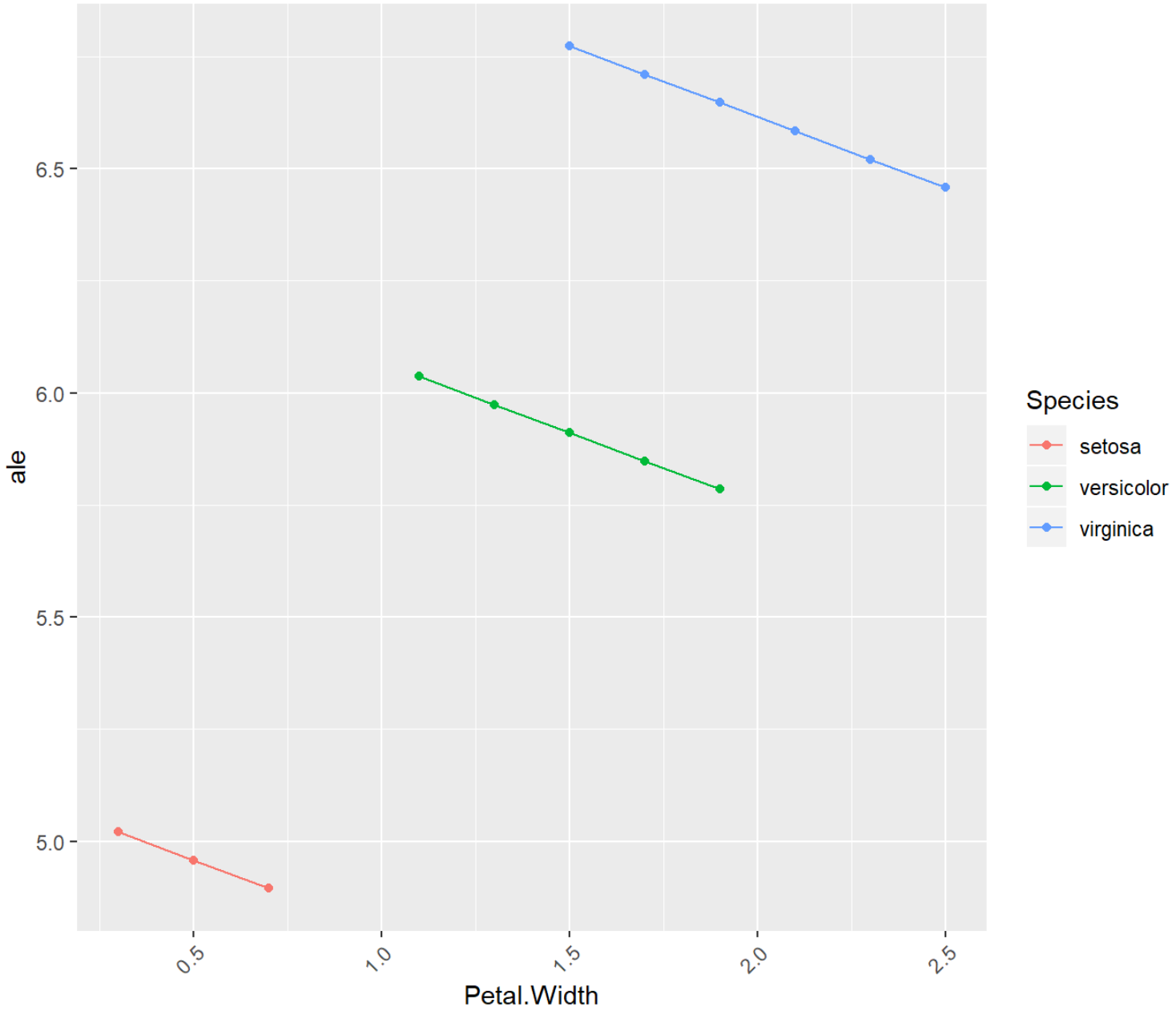
```
plot(light_profile(fl, v = "Petal.Width", by = "Species"))
```



Effects of input variables

ALE plot

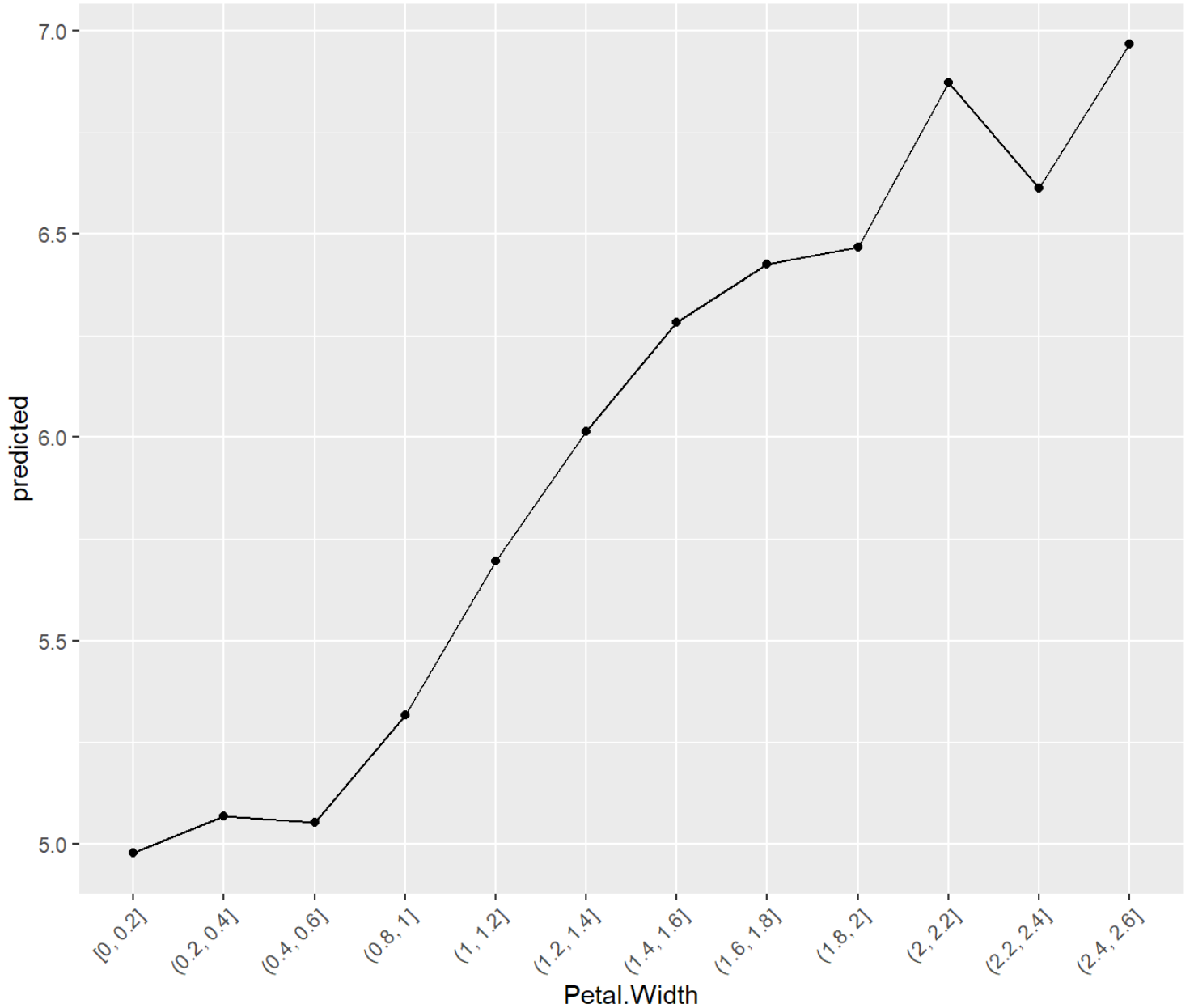
```
plot(light_profile(fl, v = "Petal.Width", by = "Species", type = "ale"))
```



Effects of input variables

Prediction plots

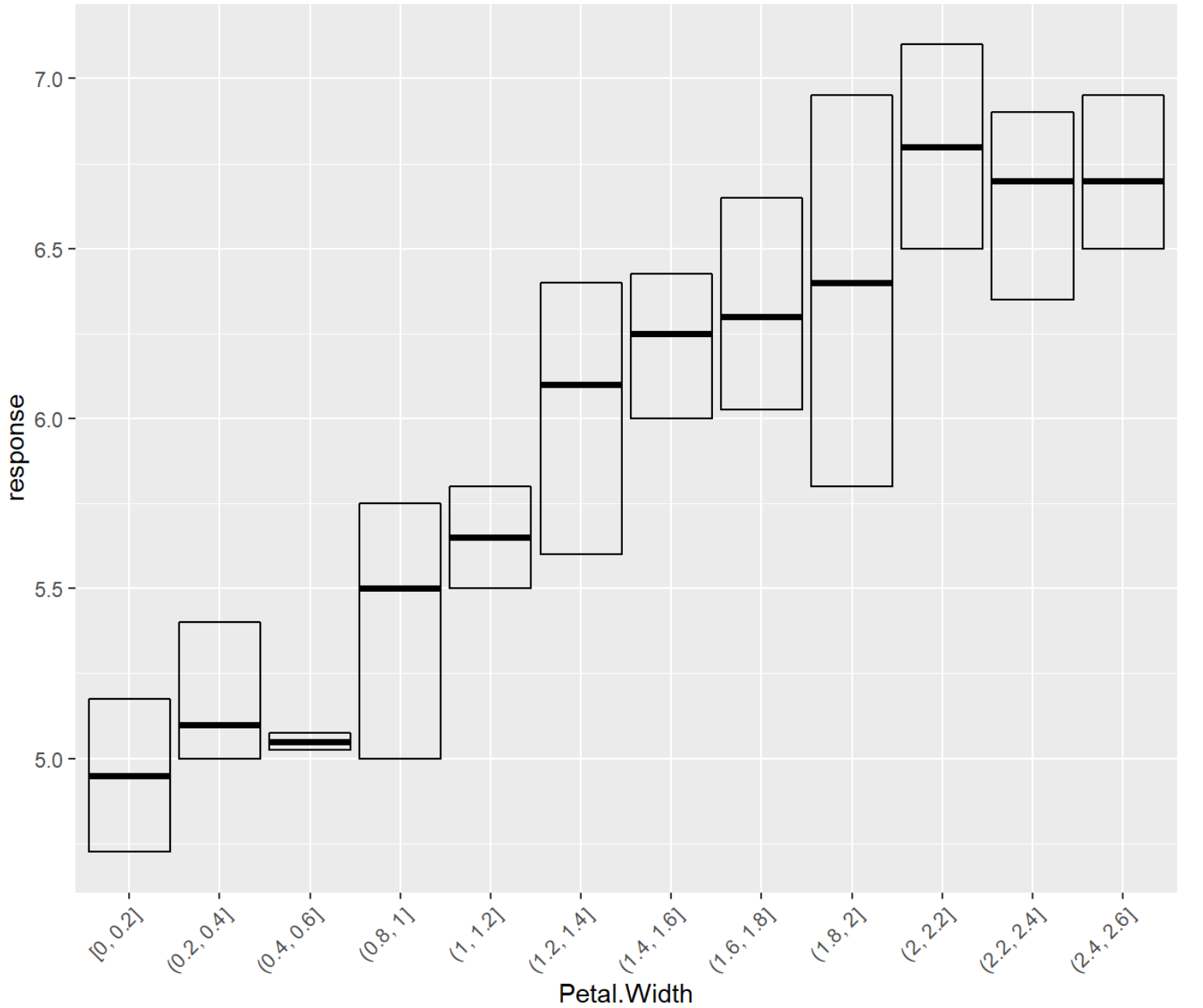
```
plot(light_profile(fl, v = "Petal.Width", type = "predicted"))
```



Effects of input variables

Response plots

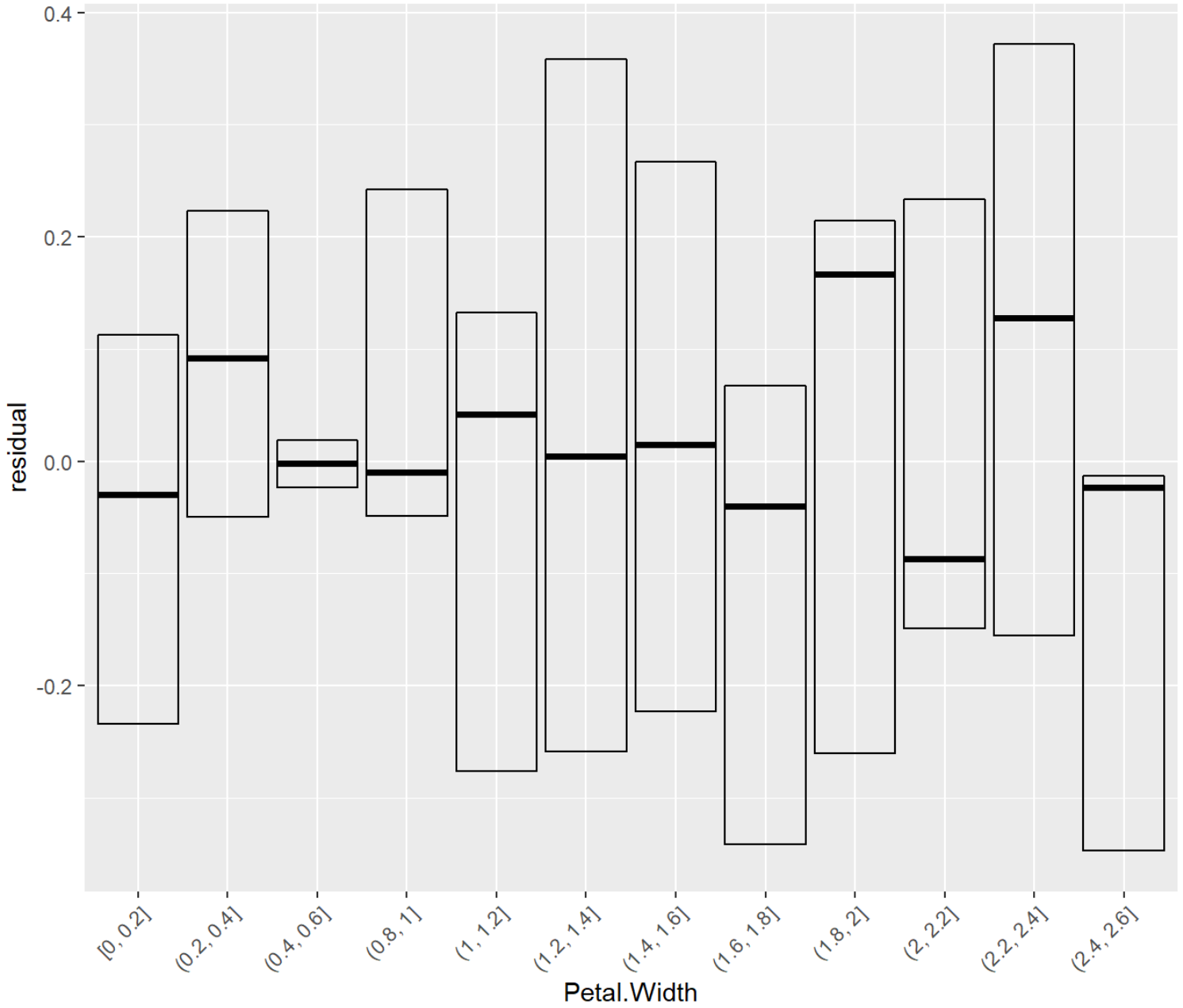
```
plot(light_profile(fl, v = "Petal.Width", type = "response", stats = "quartiles"))
```



Effects of input variables

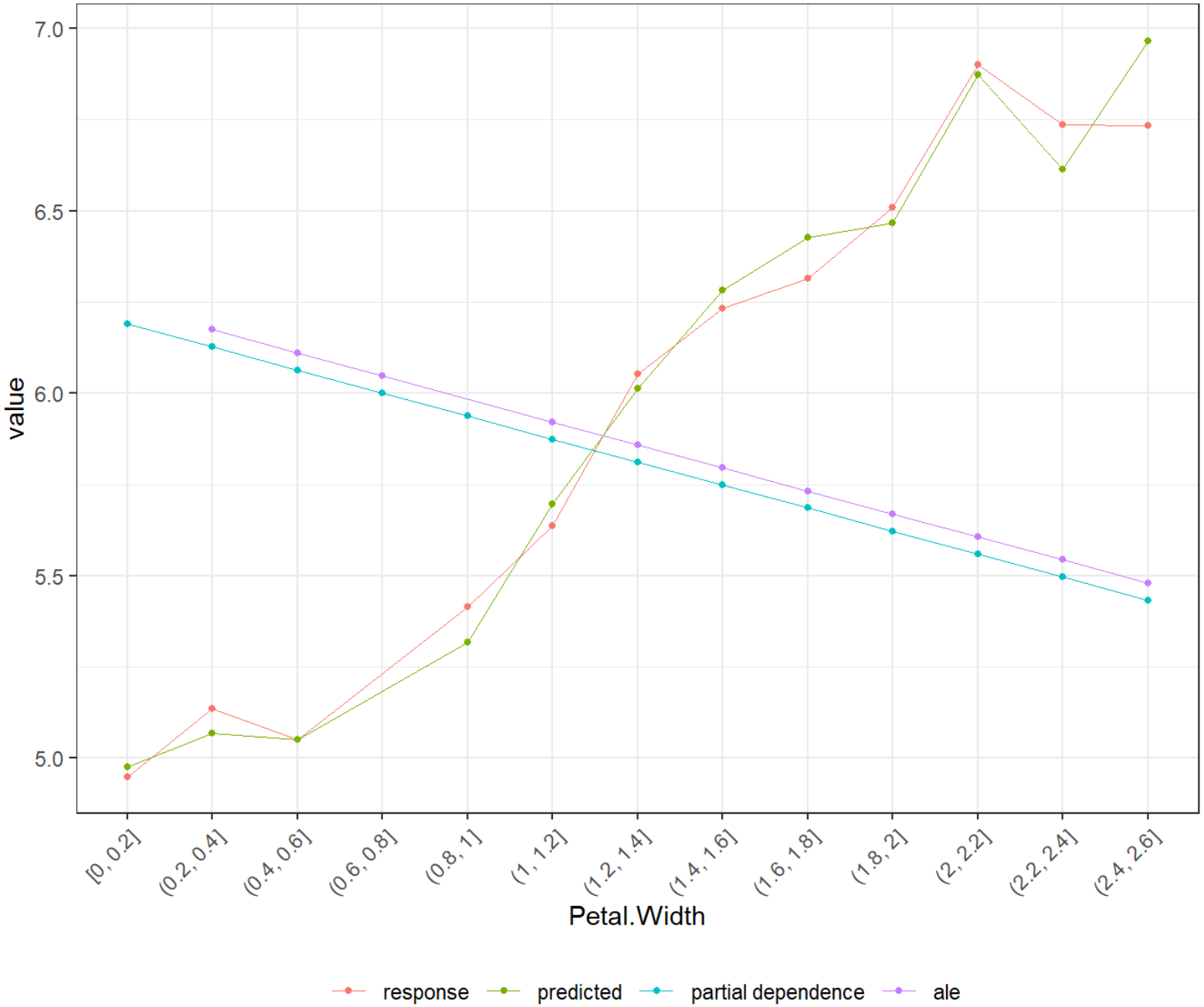
Residual plots

```
plot(light_profile(fl, v = "Petal.Width", type = "residual", stats = "quartiles"))
```



Effects of input variables

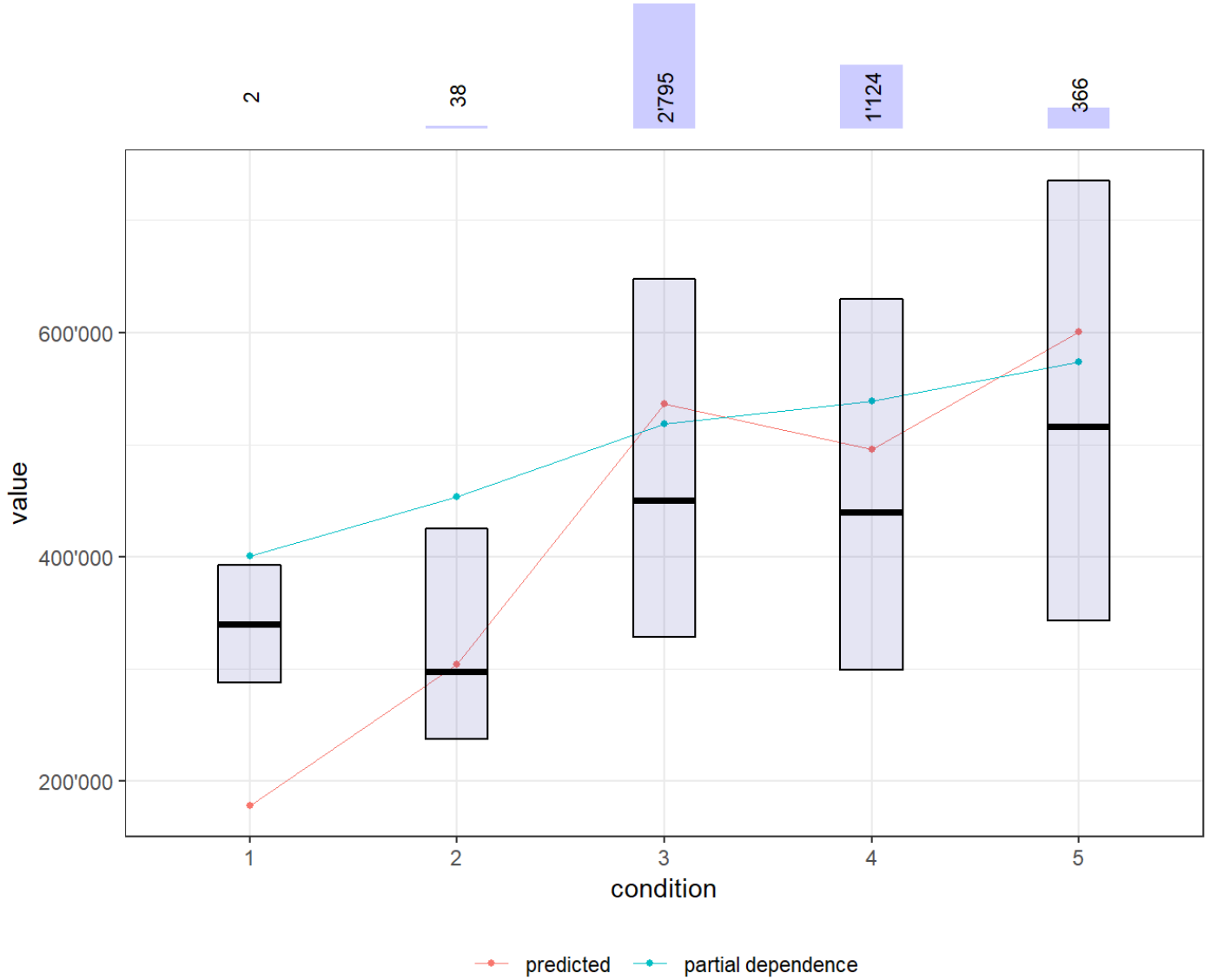
```
plot(light_effects(fl, v = "Petal.Width"), use = "all")
```



Effects of input variables

plot counts

```
eff <- light_effects(fl_lm, v = "condition", stats = "quartiles")
p <- plot(eff, rotate_x = FALSE) +
   scale_y_continuous(labels = format_y)
plot_counts(p, eff, fill = "blue", alpha = 0.2, width = 0.3)
```



Interaction strength

Besides measuring overall variable importance, an interesting aspect is to measure the strength of non-additivity associated with each covariable (i.e. the overall interaction strength) and/or between pairs of covariables.

H-statistic is a measure of total interaction strength per covariable.

How much of their variability is unexplained by the main effect?

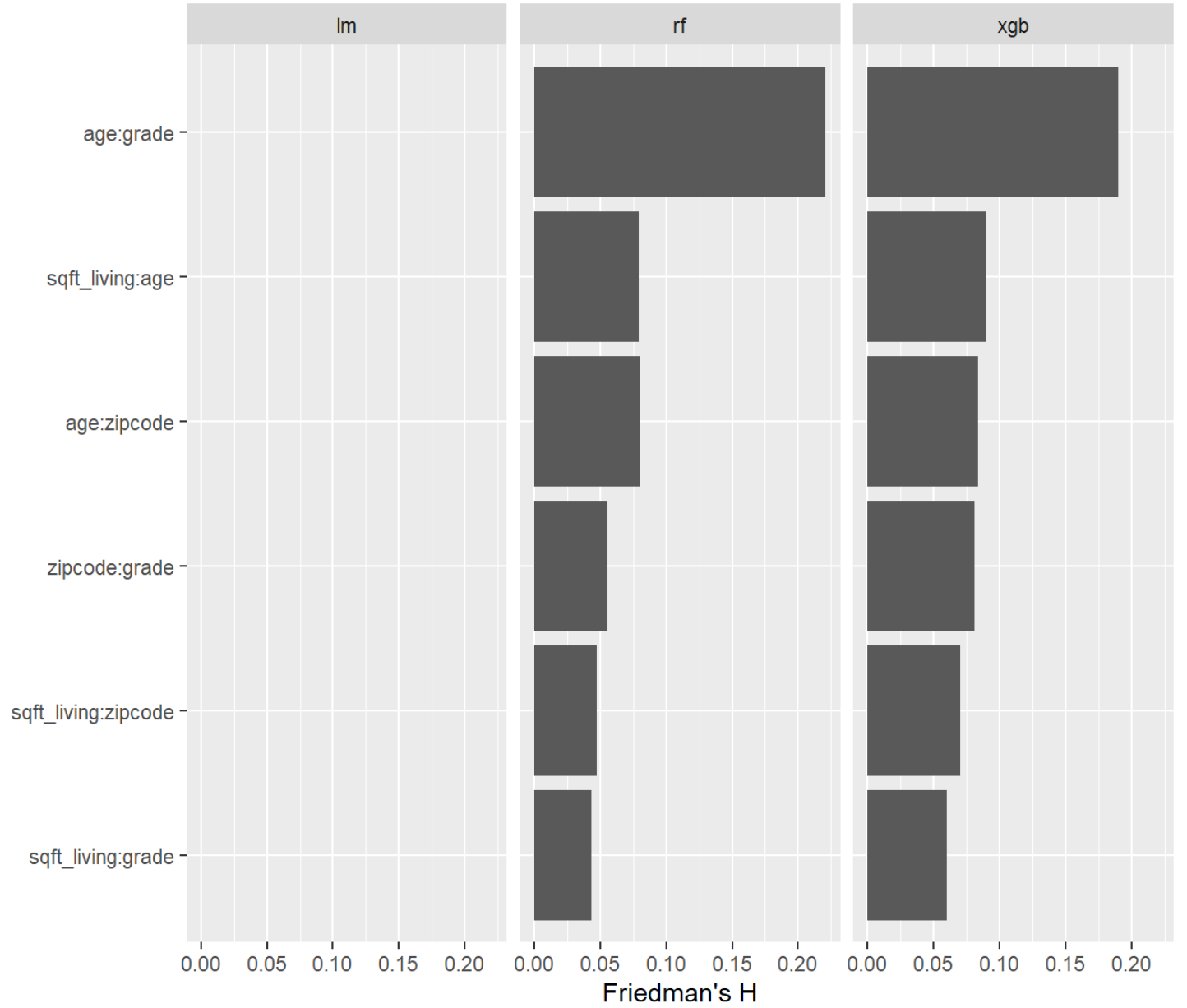
Friedman's H

based on centered partial dependence

$$H^2 = \frac{\sum\left(PD(x,z) - PD(x) - PD(z)\right)^2}{\sum PD(x,z)^2}$$

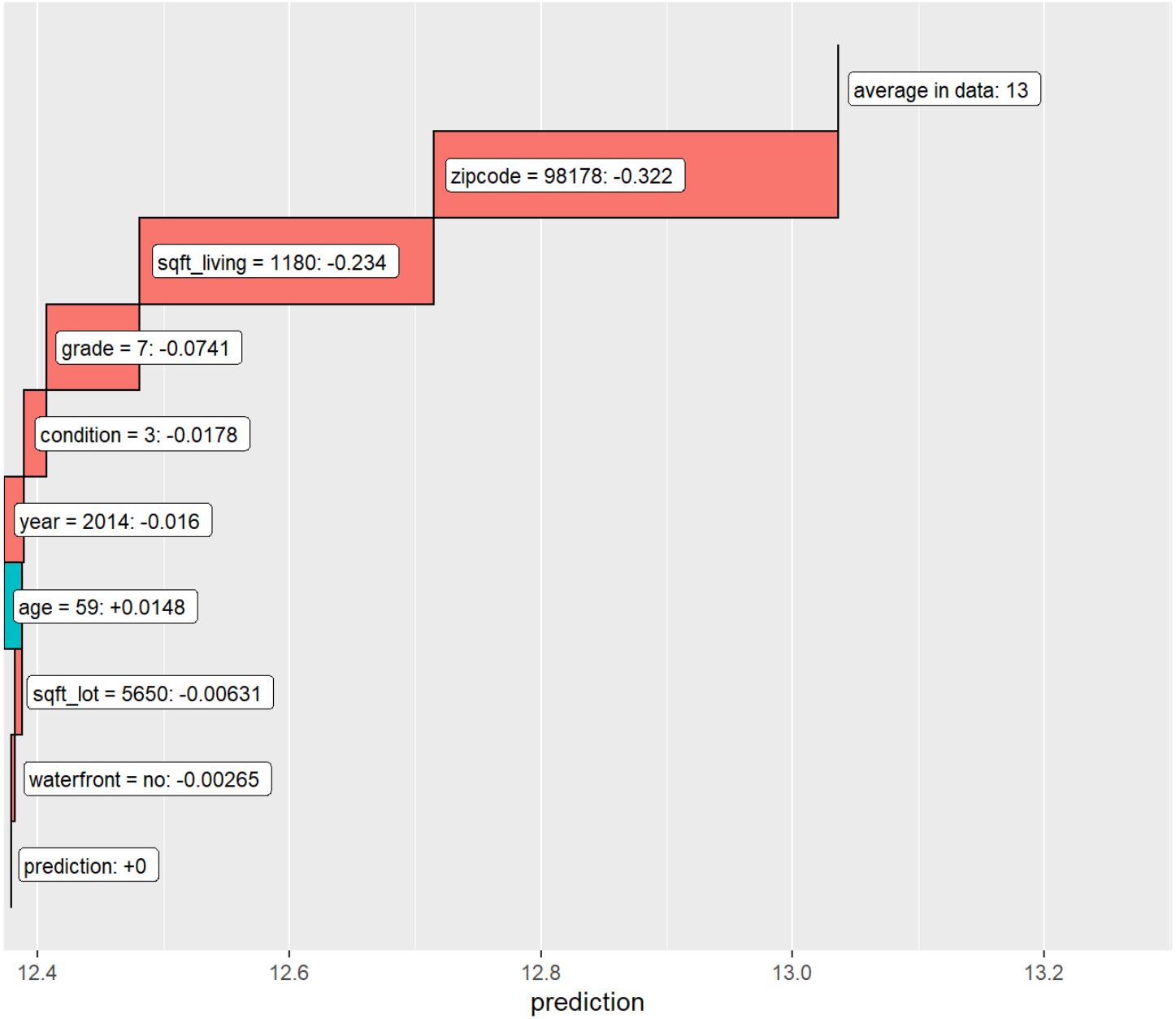
Interaction strength

```
st_pair <- light_interaction(fls, v = most_important(st, 4), pairwise = TRUE)
plot(st_pair)
```



Variable contribution breakdown for single observations

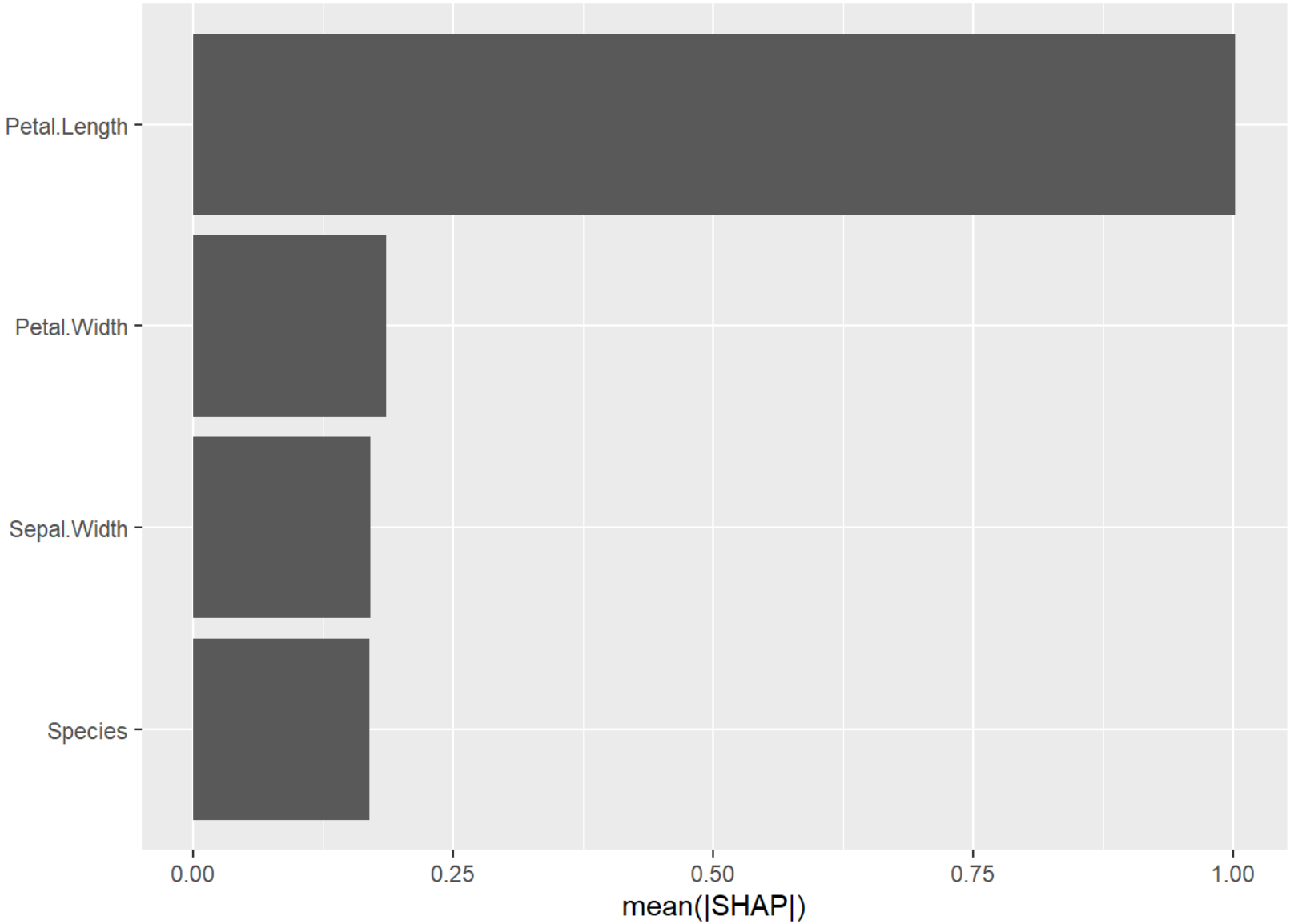
```
bd <- light_breakdown(fl_lm, new_obs = valid[1, ], v = x, n_max = 1000, seed = 74)
plot(bd, size = 3)
```



SHAP variable importance

```
fit <- lm(Sepal.Length ~ . + Petal.Length:Species, data = iris)
x <- flashlight(model = fit, label = "lm", data = iris, y = "Sepal.Length")
x <- add_shap(x, verbose = FALSE)
plot(light_importance(x, type = "shap"))
```

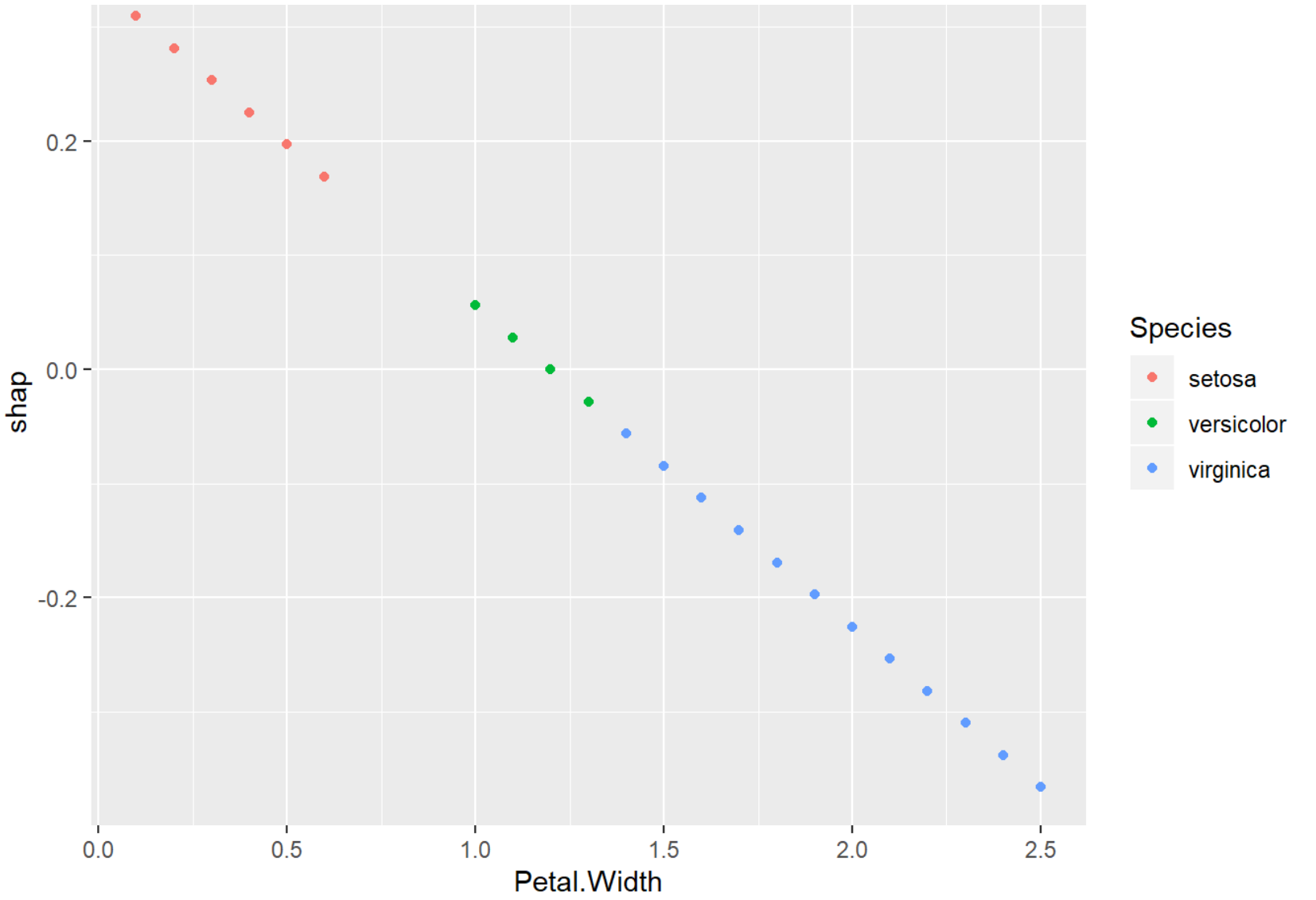
```
Coefficients:
                      (Intercept)
                           2.7562
                      Sepal.Width
                           0.5045
                     Petal.Length
                           0.4032
                      Petal.Width
                          -0.2814
                Speciesversicolor
                          -0.9654
                 Speciesvirginica
                          -2.2209
Petal.Length:Speciesversicolor
                           0.3294
 Petal.Length:Speciesvirginica
                           0.5194
```



SHAP variable contributions

```
Coefficients:
                                (Intercept)
                                     2.7562
                                Sepal.Width
                                     0.5045
                               Petal.Length
                                     0.4032
                                Petal.Width
                                    -0.2814
                           Speciesversicolor
                                    -0.9654
                            Speciesvirginica
                                    -2.2209
Petal.Length:Speciesversicolor
                                     0.3294
 Petal.Length:Speciesvirginica
                                     0.5194
```

```
fit <- lm(Sepal.Length ~ . + Petal.Length:Species, data = iris)
x <- flashlight(model = fit, label = "lm", data = iris, y = "Sepal.Length")
x <- add_shap(x, verbose = FALSE)
ls <- light_scatter(x, type = "shap", v = "Petal.Width", by=c("Species"))
plot(ls, swap_dim = T)
```



Variable contribution SHAP for single observations

```
ls <- light_scatter(x, type = "shap", v = "Petal.Length", by=c("Species"))
plot(ls, swap_dim = T)
```

```
Coefficients:
                                (Intercept)
                                     2.7562
                                Sepal.Width
                                     0.5045
                               Petal.Length
                                     0.4032
                                Petal.Width
                                    -0.2814
                          Speciesversicolor
                                    -0.9654
                           Speciesvirginica
                                    -2.2209
Petal.Length:Speciesversicolor
                                     0.3294
 Petal.Length:Speciesvirginica
                                     0.5194
```