Aspect_importance update: 22-26.7.19

Basic example

Titanic set, basic usage of aspect_importance, aspect list builded manually.

```
library("DALEX")
titanic <- na.omit(titanic)
model_titanic_glm <- glm(survived == "yes" ~ class+gender+age+sibsp+parch+fare+embarked,
                         titanic, family = "binomial")
```

```
aspects <- list(wealth = c("class", "fare"), family = c("sibsp", "parch"),
                personal = c("age","gender"), embarked = "embarked")

passenger <- data.frame(
  class = factor("1st", levels = c("1st", "2nd", "3rd", "deck crew",
                                   "engineering crew", "restaurant staff", "victualling crew")),
  gender = factor("male", levels = c("female", "male")),
  age = 8,
  sibsp = 0,
  parch = 0,
  fare = 72,
  embarked = factor("Southampton", levels = c("Belfast", "Cherbourg",
                                              "Queenstown", "Southampton"))
)

predict(model_titanic_glm, passenger)
```

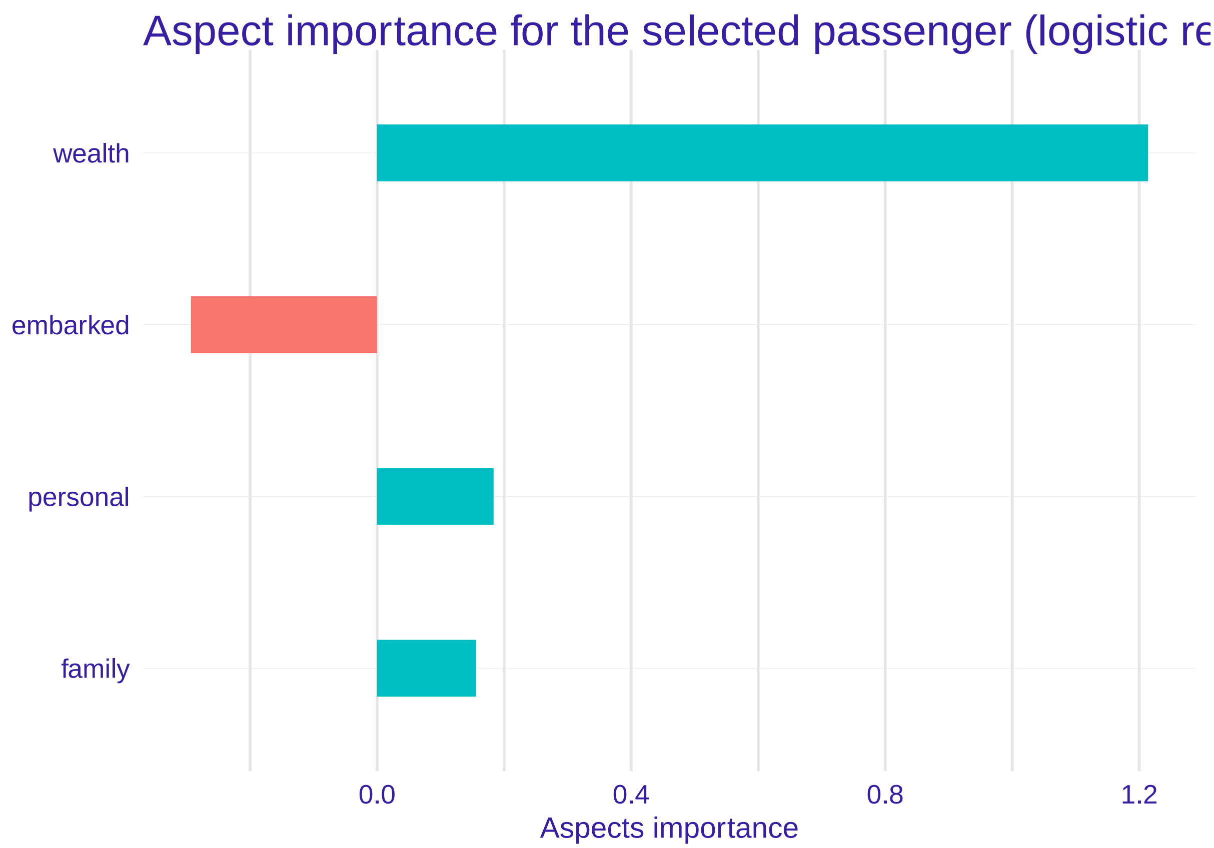
```
##         1
## 0.6724878
```

```
library("ggplot2")
library("ingredients")

titanic_glm_ai <- aspect_importance(model_titanic_glm, titanic,
                                    new_observation = passenger, aspects_list = aspects)

titanic_glm_ai
```

```
##    aspects importance
## 2   wealth  1.2137731
## 5 embarked -0.2930328
## 4 personal  0.1834507
## 3   family  0.1557833
```

Aspect **wealth** (features *class* and *fare*) has the biggest postive contribution on survival prediction for the passenger.

```
plot(titanic_glm_ai) + ggtitle("Aspect importance for the selected passenger (logistic reg.)")
```



Additional options

Boston housing dataset.

- Aspect list builded automatically (basing on pairwise correlations) with group_variables() function.
- Experiments with a plot that shows both aspect importance and correlations.
- Comparing aspect_importance() function results builded on aspect list with only single features with break down and oscillations.

```
library(mlbench)
data("BostonHousing2")
Boston <- BostonHousing2

Boston <- dplyr::select(Boston,-c(town, tract, lon, lat, medv))
Boston_numeric <- dplyr::select(Boston,-c(chas, rad))
Boston.lm2 <- lm(cmedv~., Boston_numeric)
```

```
obs_1 <- names(sort(abs(Boston.lm2$residuals),decreasing = T)[50])
obs_1 <- Boston_numeric[obs_1,]
obs_1_comparison_df <- as.data.frame(rbind(obs_1,apply(Boston_numeric,2,fivenum)))
```

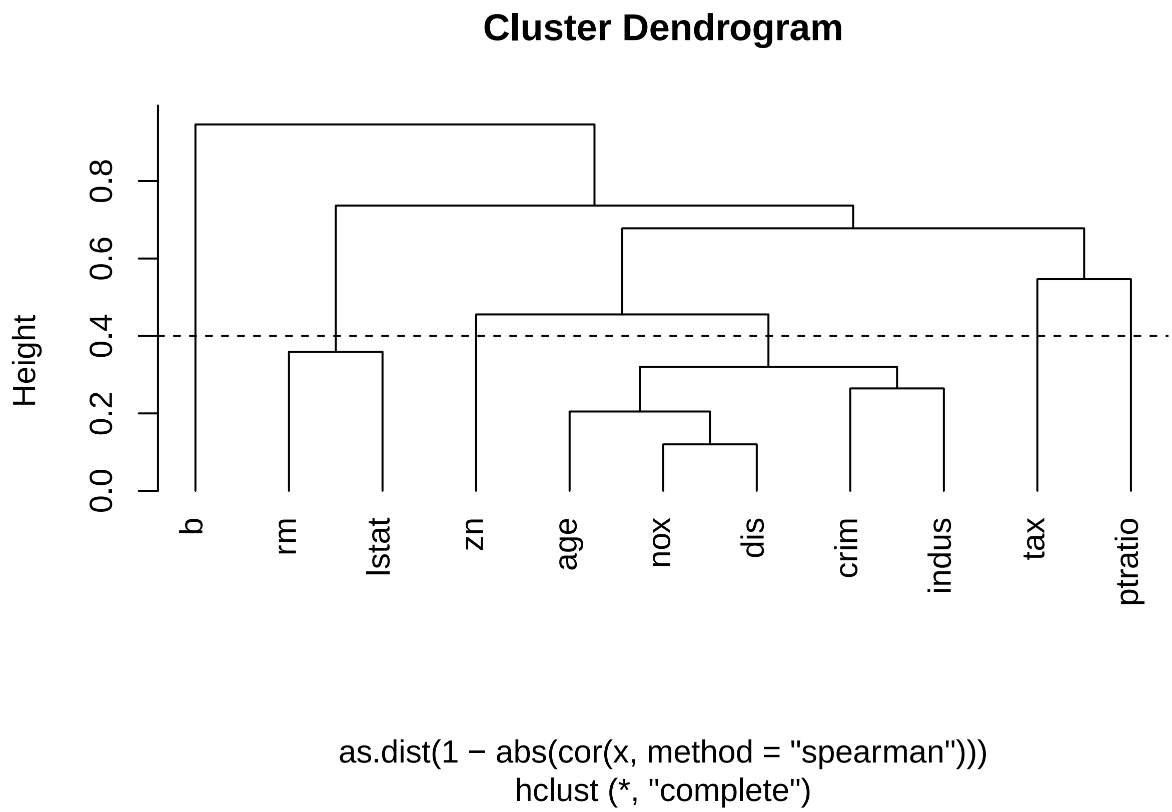
```
rownames(obs_1_comparison_df)[c(2:6)] <- c("minimum", "lower-hinge",
                                            "median", "upper-hinge", "maximum")
obs_1_comparison_df
```

```
##              cmedv     crim    zn indus   nox     rm   age      dis tax
## 257           44.0  0.01538  90.0  3.75 0.394 7.4540  34.2  6.33610 244
## minimum        5.0  0.00632   0.0  0.46 0.385 3.5610   2.9  1.12960 187
## lower-hinge   17.0  0.08199   0.0  5.19 0.449 5.8850  45.0  2.10000 279
## median        21.2  0.25651   0.0  9.69 0.538 6.2085  77.5  3.20745 330
## upper-hinge   25.0  3.67822  12.5 18.10 0.624 6.6250  94.1  5.21190 666
## maximum       50.0 88.97620 100.0 27.74 0.871 8.7800 100.0 12.12650 711
##              ptratio      b lstat
## 257            15.90 386.34  3.11
## minimum        12.60   0.32  1.73
## lower-hinge    17.40 375.33  6.93
## median         19.05 391.44 11.36
## upper-hinge    20.20 396.23 16.96
## maximum        22.00 396.90 37.97
```

```
predict(Boston.lm2, obs_1)
```

```
##      257
## 37.14531
```

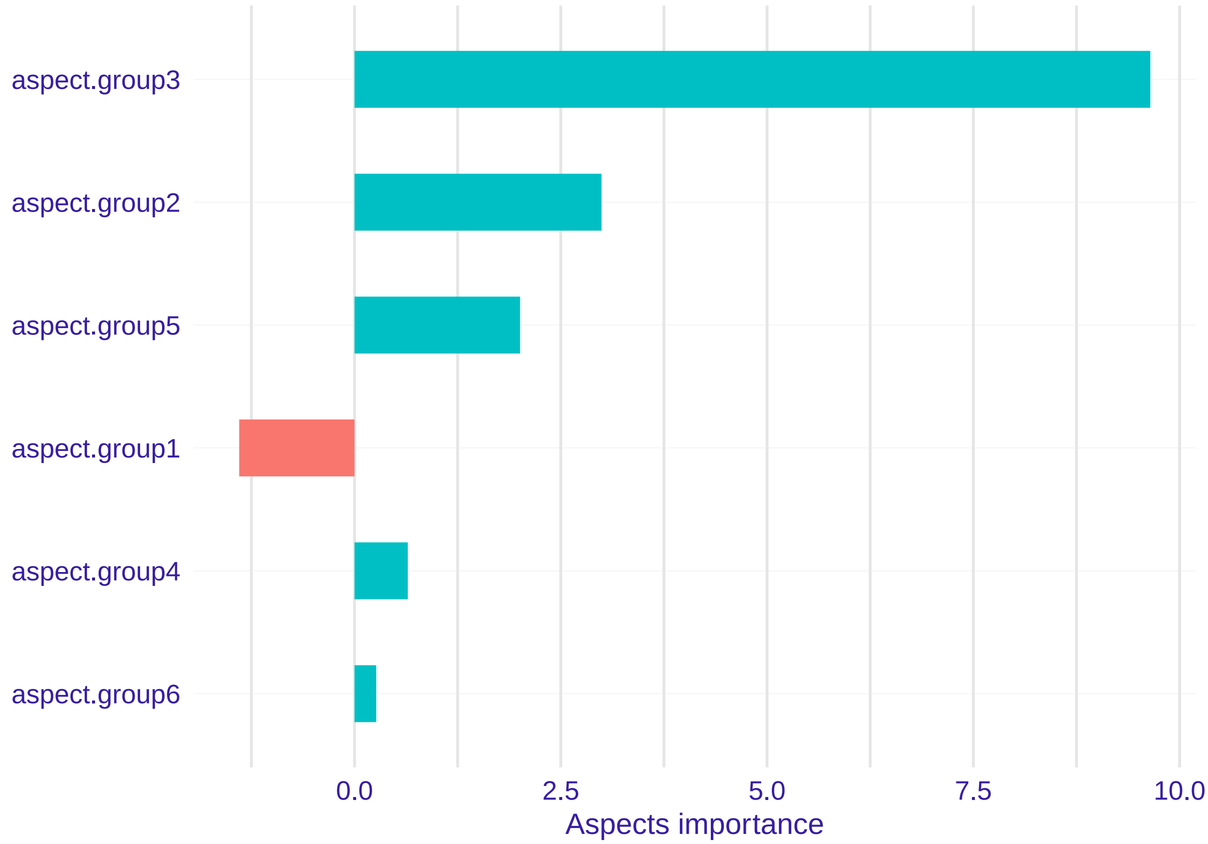
```
asp_list <- group_variables(dplyr::select(Boston_numeric,-cmedv), p = 0.6,
                            clust_method = "complete",draw_tree = T)
```



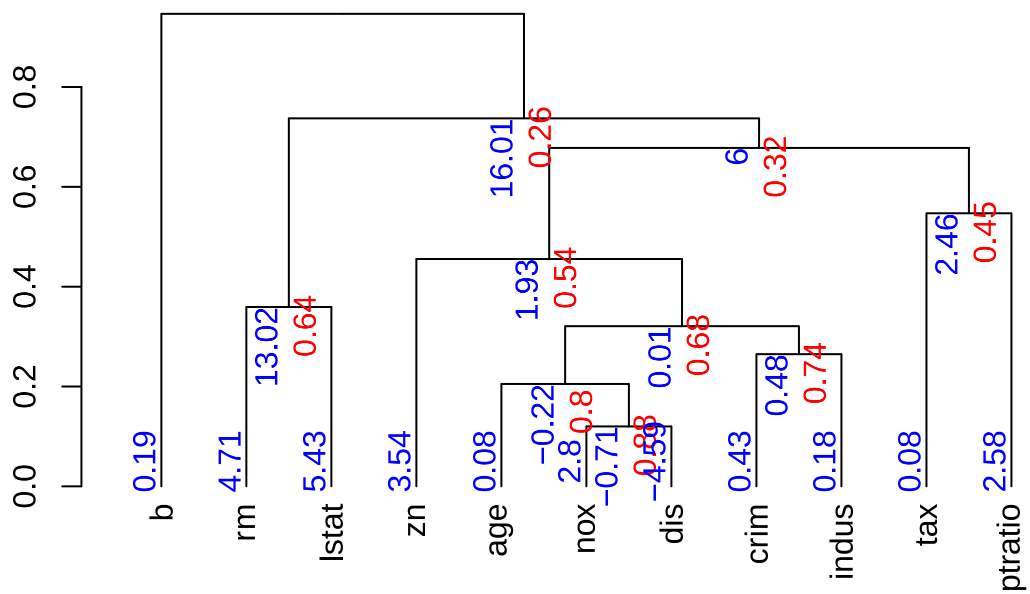

```
Boston_ai <- aspect_importance(Boston.lm2, Boston_numeric,
                               new_observation = obs_1, aspects_list = asp_list,
                               B = 500, method = "default")

add_additional_information(Boston_ai, Boston_numeric, aspect_list = asp_list,
                           show_cor = T)
```

```
##         aspects importance                   features min_cor sign
## 4 aspect.group3     9.6451                  rm, lstat  0.6408  neg
## 3 aspect.group2     2.9939                         zn      NA
## 6 aspect.group5     2.0065                    ptratio      NA
## 2 aspect.group1    -1.3975 crim, indus, nox, age, dis  0.6795  neg
## 5 aspect.group4     0.6460                        tax      NA
## 7 aspect.group6     0.2633                          b      NA
```

```
plot(Boston_ai)
```



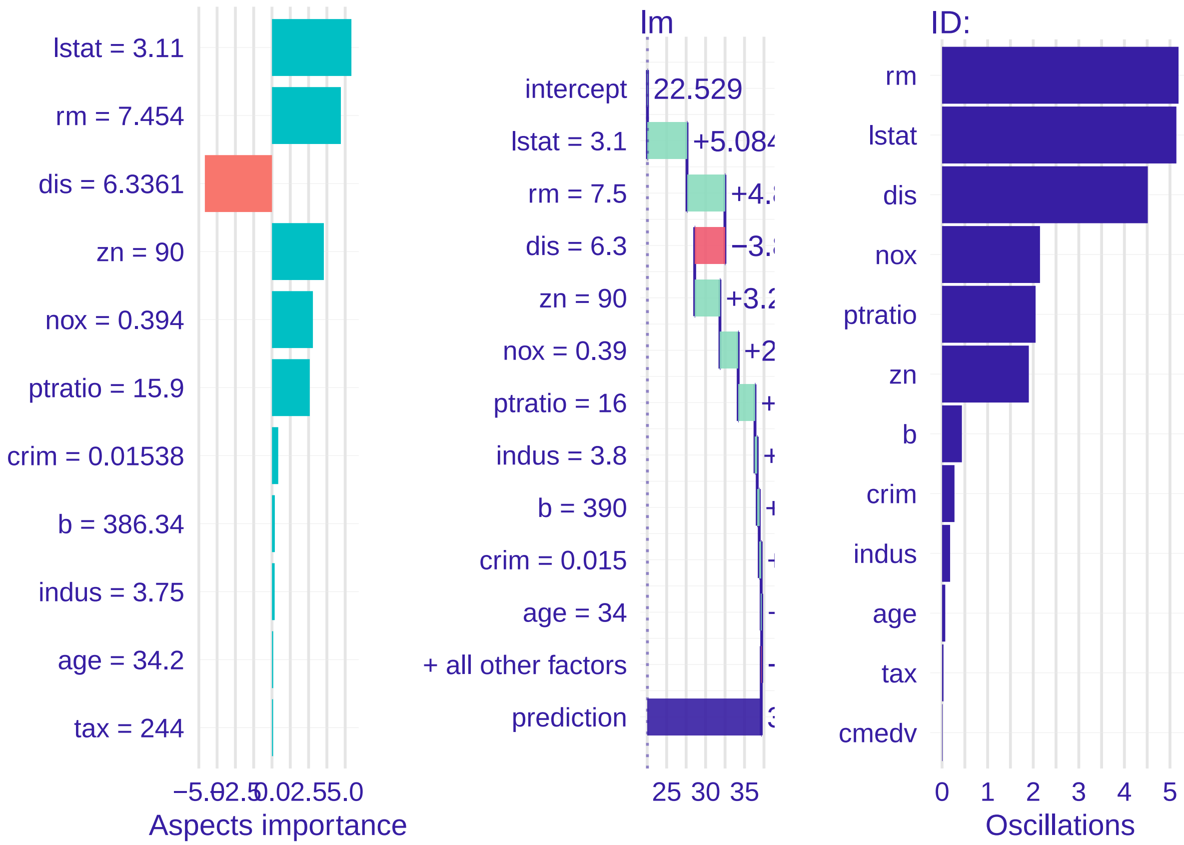

```
draw_plot_with_labels(dplyr::select(Boston_numeric,-cmedv), Boston.lm2,
                      Boston_numeric, new_observation = obs_1, B = 1000, dendtree_text_vert = TRUE)
```




```
## [1] "T"
```

```r
library(gridExtra)
library(iBreakDown)

aspect_importance_boston_glm <- aspect_importance_single(Boston.lm2,
                                                          Boston_numeric, predict,
                                                          new_observation = dplyr::select(obs_1,-cmedv),
                                                          B = 1000)
explainer_boston_lm <- DALEX::explain(Boston.lm2,
                         Boston_numeric,
                         y = Boston_numeric$cmedv)

bd_boston_glm <- break_down(explainer_boston_lm,
                        new_observation = obs_1)

cp_boston_lm <- ceteris_paribus(explainer_boston_lm, obs_1)
oscillations_boston_lm <- calculate_oscillations(cp_boston_lm)[,c(1,3)]

# aspect_importance_boston_glm
# bd_boston_glm
# oscillations_boston_lm

p1 <- plot(aspect_importance_boston_glm)
p2 <- plot(bd_boston_glm)
```

```r
p3 <- plot(oscillations_boston_lm)

grid.arrange(p1, p2, p3, nrow = 1)
```




```r
bd_boston_glm <- bd_boston_glm[-c(1,nrow(bd_boston_glm)-1,nrow(bd_boston_glm)),c(1:2)]
oscillations_boston_lm <- oscillations_boston_lm[-nrow(oscillations_boston_lm),]

cbind(aspect_importance_boston_glm, bd_boston_glm,
           oscillations_boston_lm)
```

```
##           aspects  importance     variable contribution _vname_
## 12   lstat = 3.11  5.42543063  lstat = 3.1   5.08360214      rm
## 6      rm = 7.454  4.70568728     rm = 7.5   4.87104251   lstat
## 8    dis = 6.3361 -4.58683426    dis = 6.3  -3.88533935     dis
## 3         zn = 90  3.54203607      zn = 90   3.24318095     nox
## 5     nox = 0.394  2.79597440   nox = 0.39   2.34194365 ptratio
## 10 ptratio = 15.9  2.58034204 ptratio = 16   2.16162089      zn
## 2  crim = 0.01538  0.42755191  indus = 3.8   0.28578025       b
## 11     b = 386.34  0.19288457      b = 390   0.26006056    crim
## 4    indus = 3.75  0.18248967 crim = 0.015   0.24681320   indus
## 7      age = 34.2  0.08164216     age = 34   0.06701450     age
## 9       tax = 244  0.07527467    tax = 240  -0.05926751     tax
##    oscillations
## 12   5.18798559
## 6    5.14023258
```

```
## 8     4.51292179
## 3     2.14871271
## 5     2.05215947
## 10    1.90335354
## 2     0.43525733
## 11    0.27293638
## 4     0.17678851
## 7     0.07060338
## 9     0.03120708
```