regressions

Gustav Idun Sloth

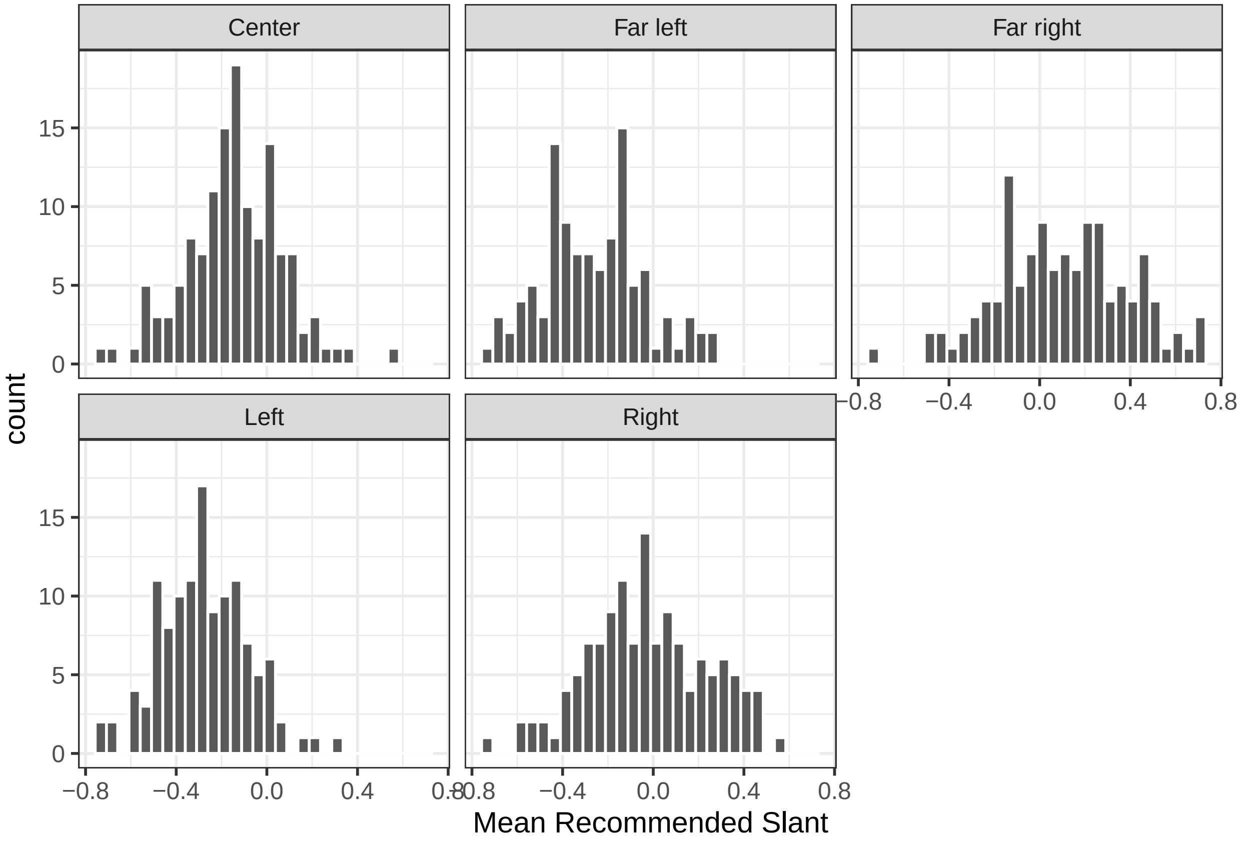
2025-11-24

```r
# import the puppet data
df <- read_csv("../../data/puppets/processed/27-12/wide.csv")
```

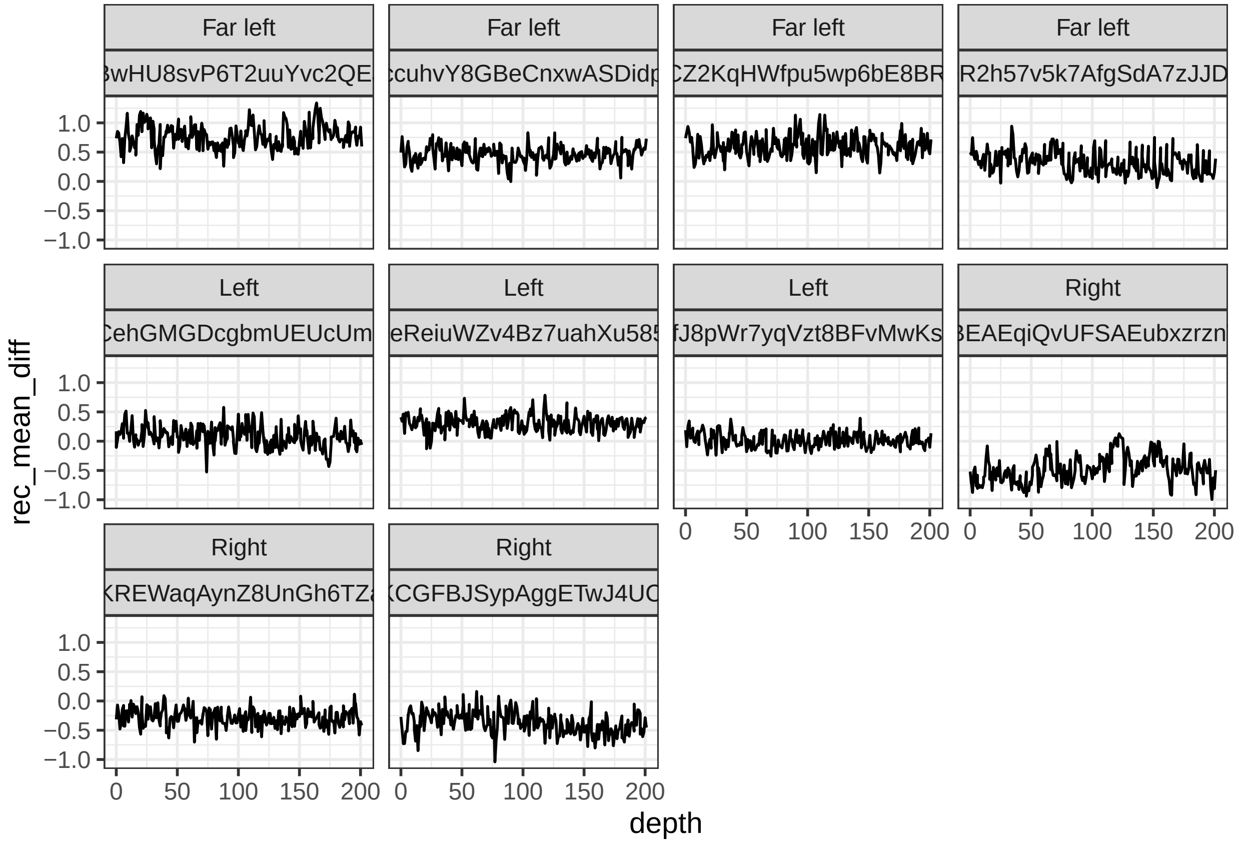
```
## New names:
## Rows: 582472 Columns: 34
## -- Column specification
## ---------------------------------------------------------- Delimiter: "," chr
## (16): puppet_id, puppet_state, puppet_cond, video_id, recs_id, recs_slan... dbl
## (16): ...1, puppet_slant, depth, video_slant, watch_time, mean_rec_slant... lgl
## (2): utility, static
## i Use ‘spec()‘ to retrieve the full column specification for this data. i
## Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
## * ‘‘ -> ‘...1‘
```

```r
# we are only interested in data during the "drifting phase"
df <- df %>% filter(puppet_state == "drifting")

df$puppet_cond <- relevel(factor(df$puppet_cond), ref="(0, 0)")
```

```r
# we plot MRS to get a sense of the distribution. Theoretically, it should be normally distributed due
ggplot(filter(df, cond_group == "random" & depth == 200), aes(x = mean_rec_slant)) +
  geom_histogram(bins = 30, color = "white") +
  facet_wrap(~ slant_label) +
  my_theme +
  labs(x="Mean Recommended Slant")
```



```
#let's look at a small sample of the raw data for the random static condition.
subids <- df %>%
  filter(cond_group == "random") %>%
  pull(puppet_id) %>%
  unique() %>%
  head(10)

p_sub <- df %>% filter(puppet_id %in% subids)

ggplot(p_sub, aes(depth, rec_mean_diff)) +
  geom_line() +
  facet_wrap(slant_label ~ puppet_id) +
  my_theme
```



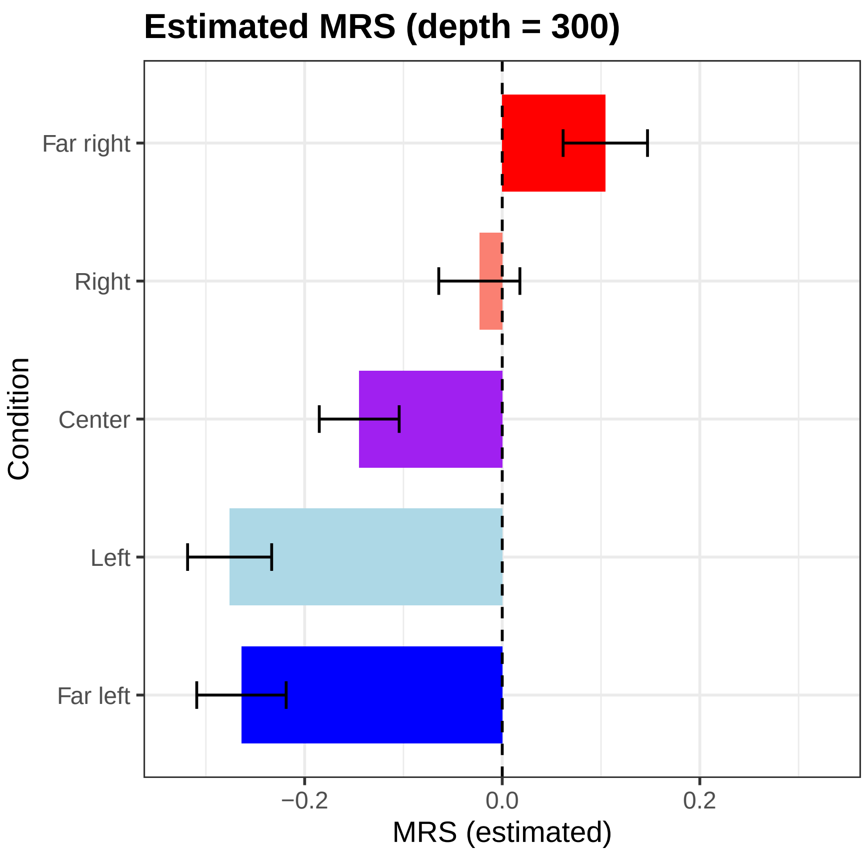
```r
#base model
md_bas <- lm(mean_rec_slant ~ slant_label, data=filter(df, cond_group == "random", depth == 200)) # we
summary(md_bas)
```

```
##
## Call:
## lm(formula = mean_rec_slant ~ slant_label, data = filter(df,
##     cond_group == "random", depth == 200))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.81424 -0.16123 -0.00808  0.15156  0.69159
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -0.14478    0.02059  -7.031 5.53e-12 ***
## slant_labelFar left   -0.11922    0.03090  -3.858 0.000127 ***
## slant_labelFar right   0.24913    0.02996   8.316 5.98e-16 ***
## slant_labelLeft       -0.13117    0.02989  -4.388 1.35e-05 ***
## slant_labelRight       0.12159    0.02934   4.144 3.90e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2384 on 607 degrees of freedom
## Multiple R-squared:  0.266,  Adjusted R-squared:  0.2611
## F-statistic: 54.99 on 4 and 607 DF,  p-value: < 2.2e-16
```

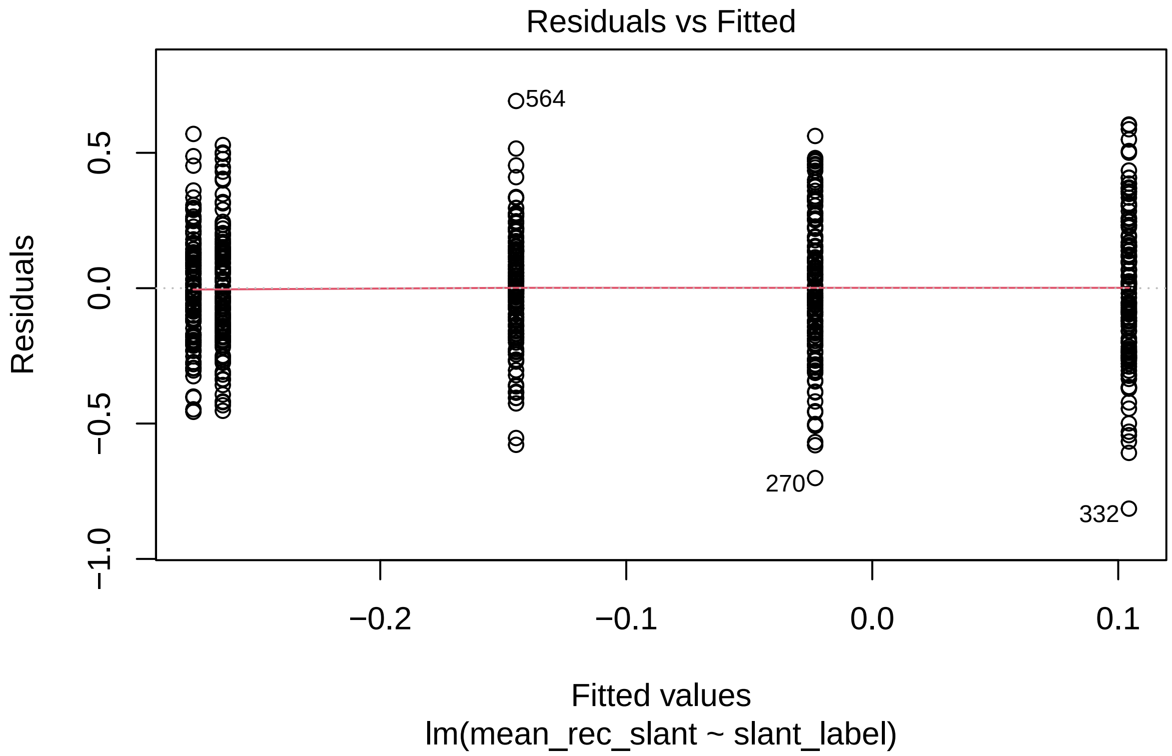
```r
#calculate marginal means
emm <- emmeans(md_bas, ~ slant_label)
summary(emm)
```

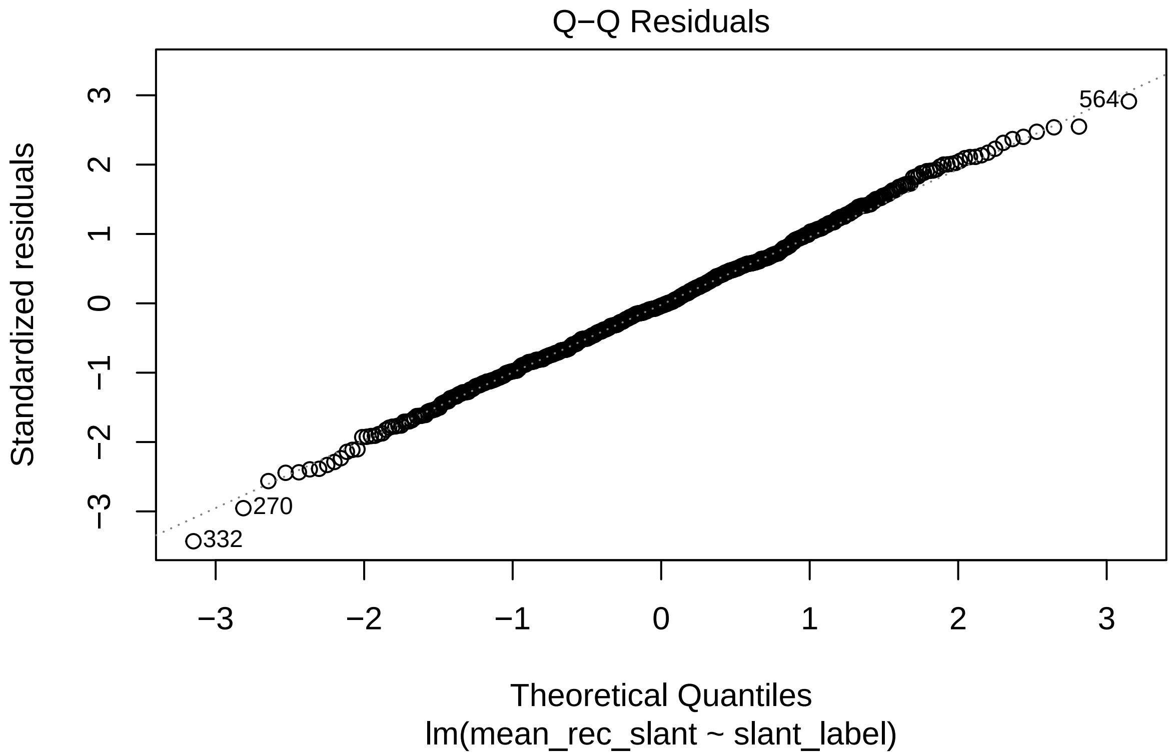
```
##  slant_label  emmean     SE  df lower.CL upper.CL
##  Center      -0.1448 0.0206 607  -0.1852  -0.1043
##  Far left    -0.2640 0.0230 607  -0.3093  -0.2187
##  Far right    0.1043 0.0218 607   0.0616   0.1471
##  Left        -0.2760 0.0217 607  -0.3185  -0.2334
##  Right       -0.0232 0.0209 607  -0.0643   0.0179
## 
## Confidence level used: 0.95
```

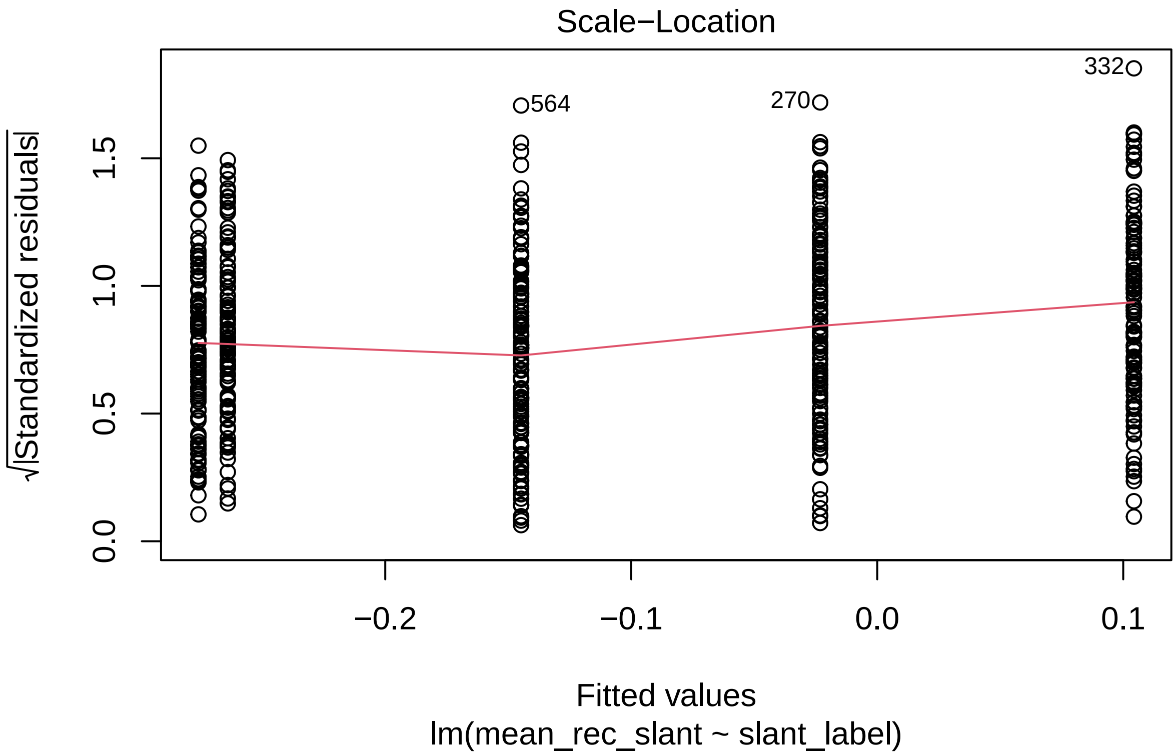
```r
# define slant scale colors according to Haroon et al (2023)
slant_label_scale = c(
  "Far left" = "blue",
  "Left" = "lightblue",
  "Center" = "purple",
  "Right" = "salmon",
  "Far right" = "red"
)

df_emm <- as.data.frame(emm)

# enforce correct order of labels
df_emm$slant_label <- factor(
  df_emm$slant_label,
  levels = c("Far left", "Left", "Center", "Right", "Far right")
)

# plot
ggplot(df_emm, aes(x = slant_label, y = emmean, fill=slant_label)) +
  geom_col(position = position_dodge(width = 0.8),
           width = 0.7) +
  geom_errorbar(
    aes(ymin = lower.CL, ymax = upper.CL),
    position = position_dodge(width = 0.8),
    width = 0.2
  ) +
  scale_y_continuous(limits = c(-0.33, 0.33)) +
  coord_flip() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  scale_fill_manual(values = slant_label_scale) +
  labs(
    title = "Estimated MRS (depth = 300)",
    x = "Condition",
    y = "MRS (estimated)"
  ) +
  my_theme +
  theme(legend.position = "none", aspect.ratio=1)
```

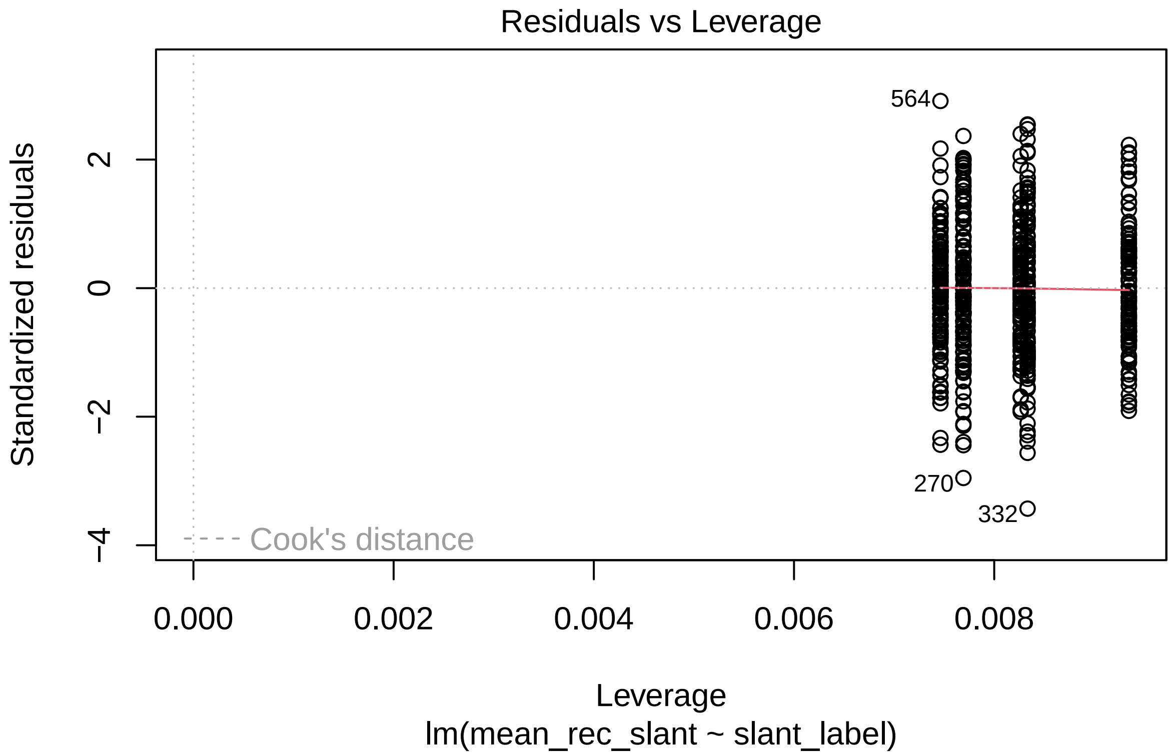


```
#quick diagnostics
plot(md_bas)
```


Residuals vs Fitted
Residuals
0.5
0.0
−0.5
−1.0
−0.2
−0.1
0.0
0.1
564
270
332
Fitted values
lm(mean_rec_slant ~ slant_label)


Q−Q Residuals
Standardized residuals
3
2
1
0
−1
−2
−3
564
270
332
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(mean_rec_slant ~ slant_label)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
−0.2
−0.1
0.0
0.1
564
270
332
Fitted values
lm(mean_rec_slant ~ slant_label)


Residuals vs Leverage
Standardized residuals
2
0
−2
−4
Leverage
lm(mean_rec_slant ~ slant_label)
0.000
0.002
0.004
0.006
0.008
564
270
332
Cook's distance