a— title: “hw1-Zhongheng Yang” output: pdf_document —

To automatically wrap the lines in output PDF:

```
library(knitr)
opts_chunk$set(tidy.opts=list(width.cutoff=25),tidy=TRUE)
```

---

prob1 choose: cty cyl(mutate to categorical cyl2) fl

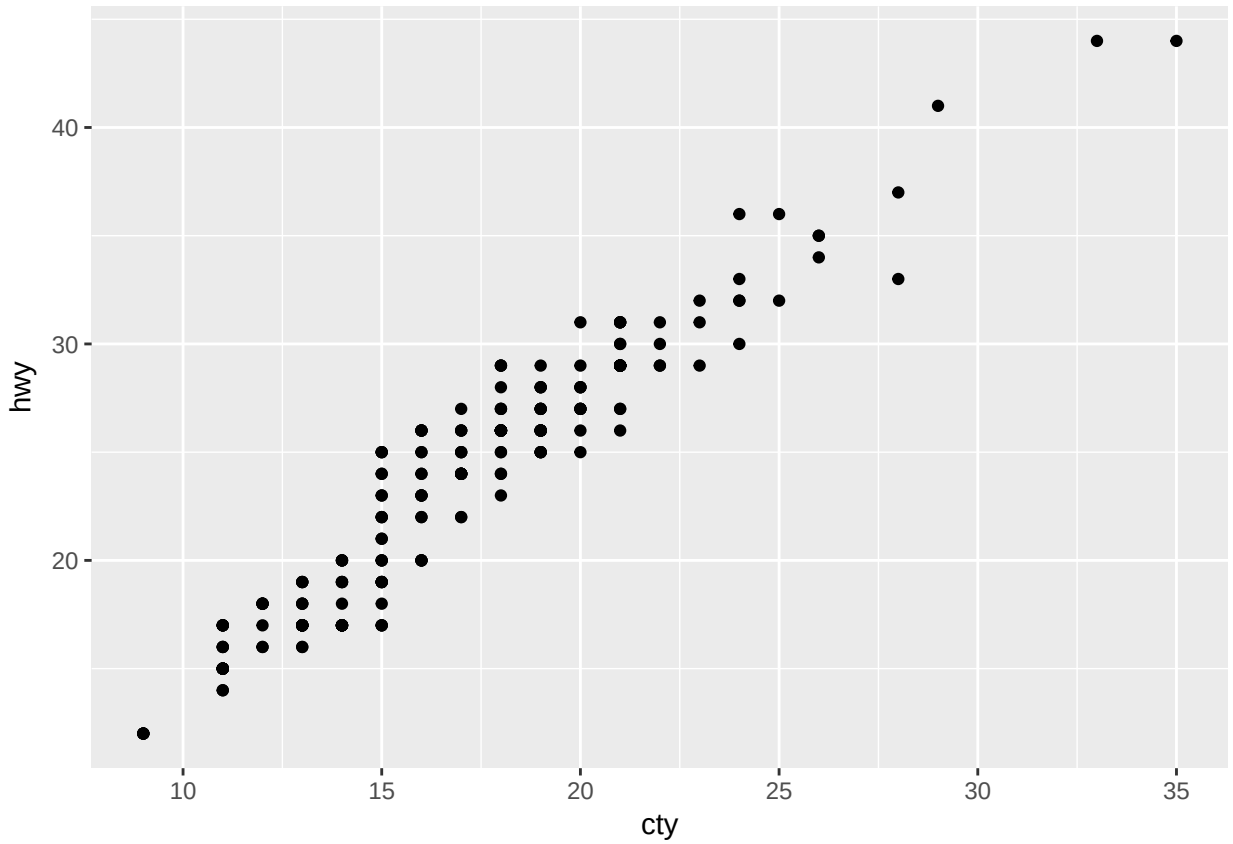
```
library(tidyverse)
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

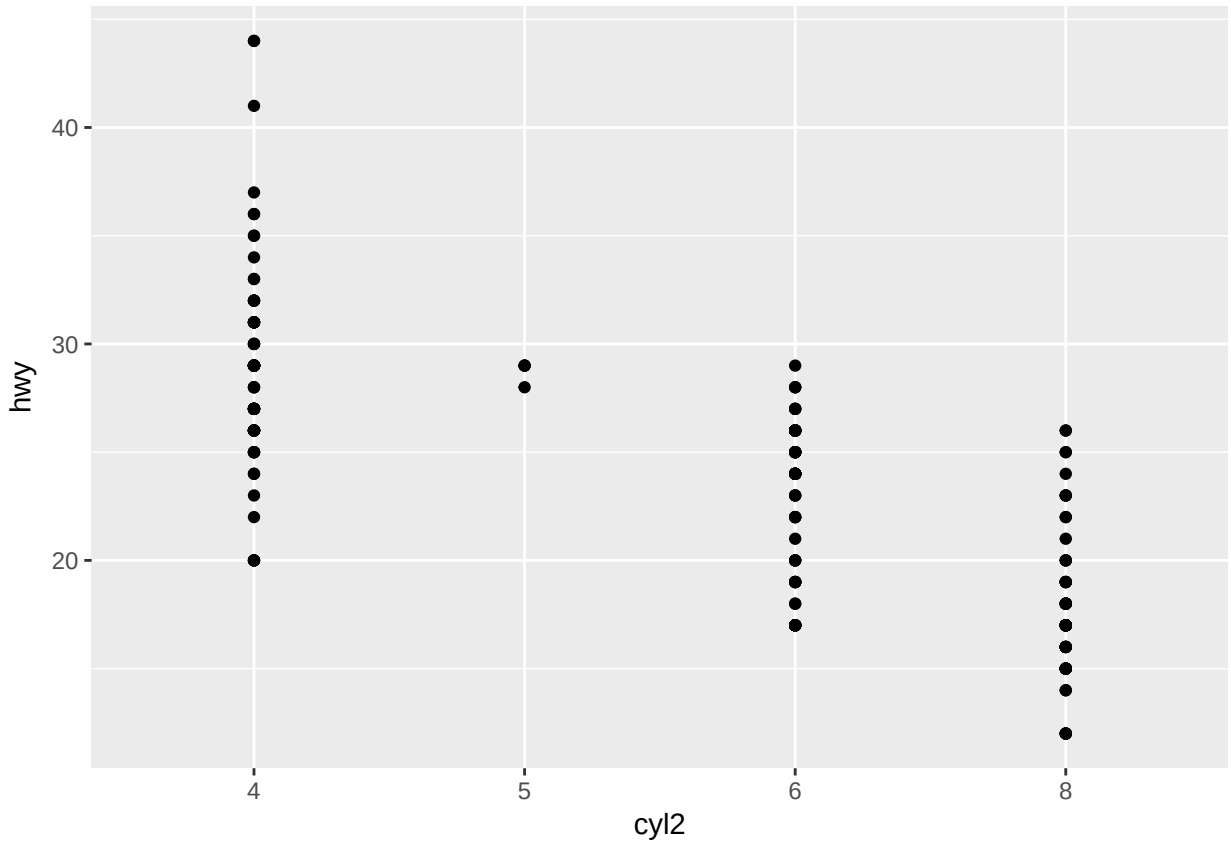
```
## Conflicts with tidy packages ----------------------------------------------
```

```
## filter(): dplyr, stats
## lag():    dplyr, stats
```

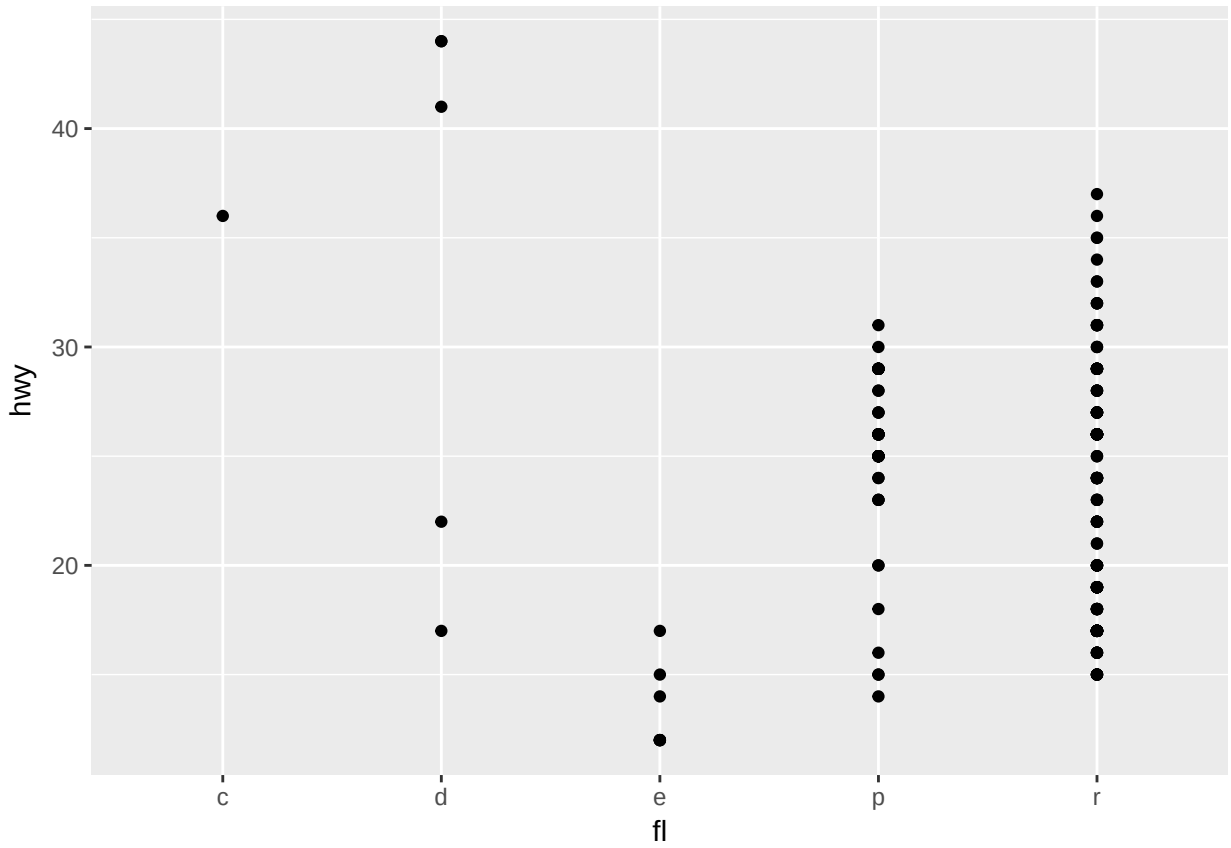
```
library(modelr)
mpgx <- transmute(mpg, cyl2 = as.factor(cyl),
    year = as.factor(year),
    trans = as.factor(trans),
    drv = as.factor(drv),
    fl = factor(fl), class = as.factor(class),
    cty = cty, hwy = hwy,
    displ = displ, )
ggplot(data = mpgx) + geom_point(aes(x = cty,
    y = hwy))
```




```
ggplot(data = mpgx) + geom_point(aes(x = cyl2,
    y = hwy))
```




```
ggplot(data = mpgx) + geom_point(aes(x = fl,
    y = hwy))
```




```
fit_mpg <- lm(hwy ~ cty +
    cyl2 + fl, data = mpgx)
fit_mpg
```

```
##
## Call:
## lm(formula = hwy ~ cty + cyl2 + fl, data = mpgx)
##
## Coefficients:
## (Intercept)          cty        cyl25        cyl26        cyl28
##      2.4911       1.3962       1.0392       0.8066       0.7641
##         fld          fle          flp          flr
##     -4.9481      -3.6235      -1.9607      -3.4025
```

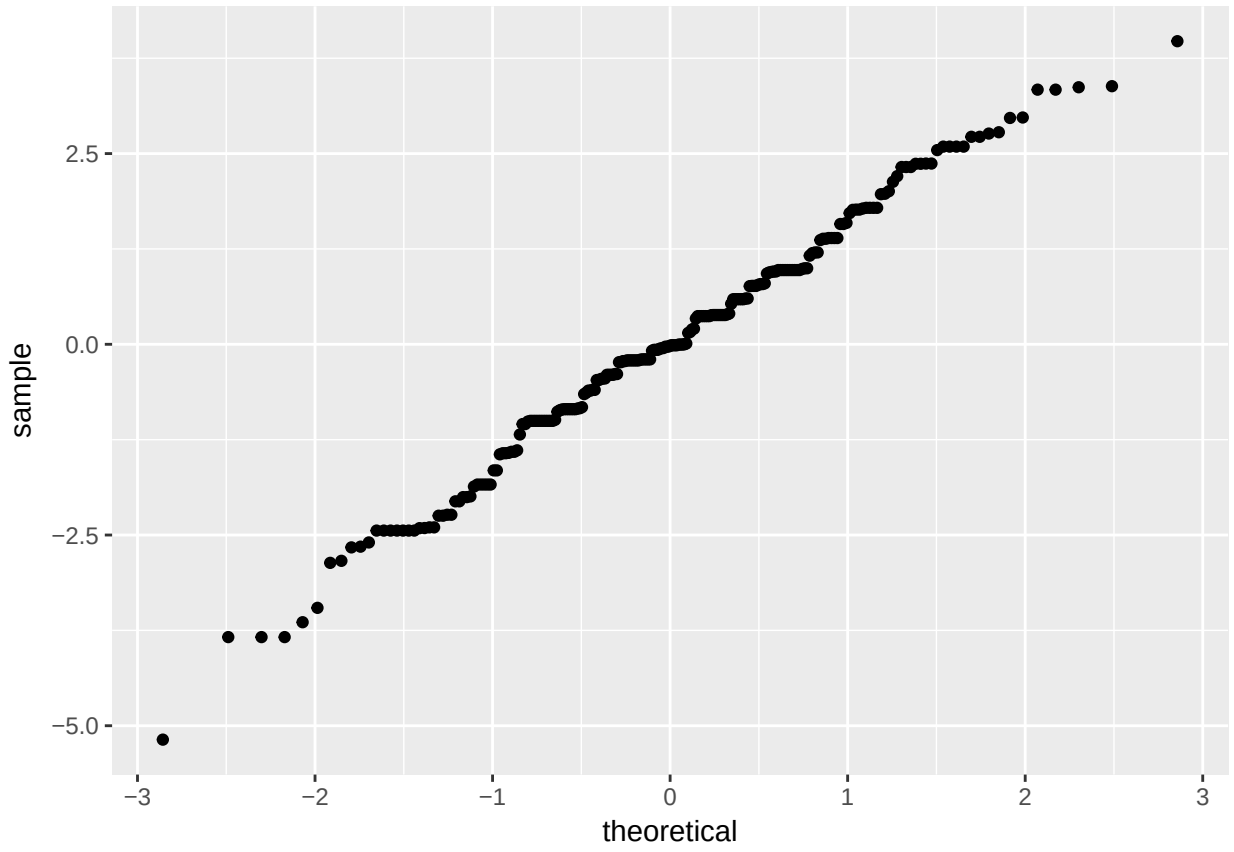
---

prob2.1 plot residual on used varable

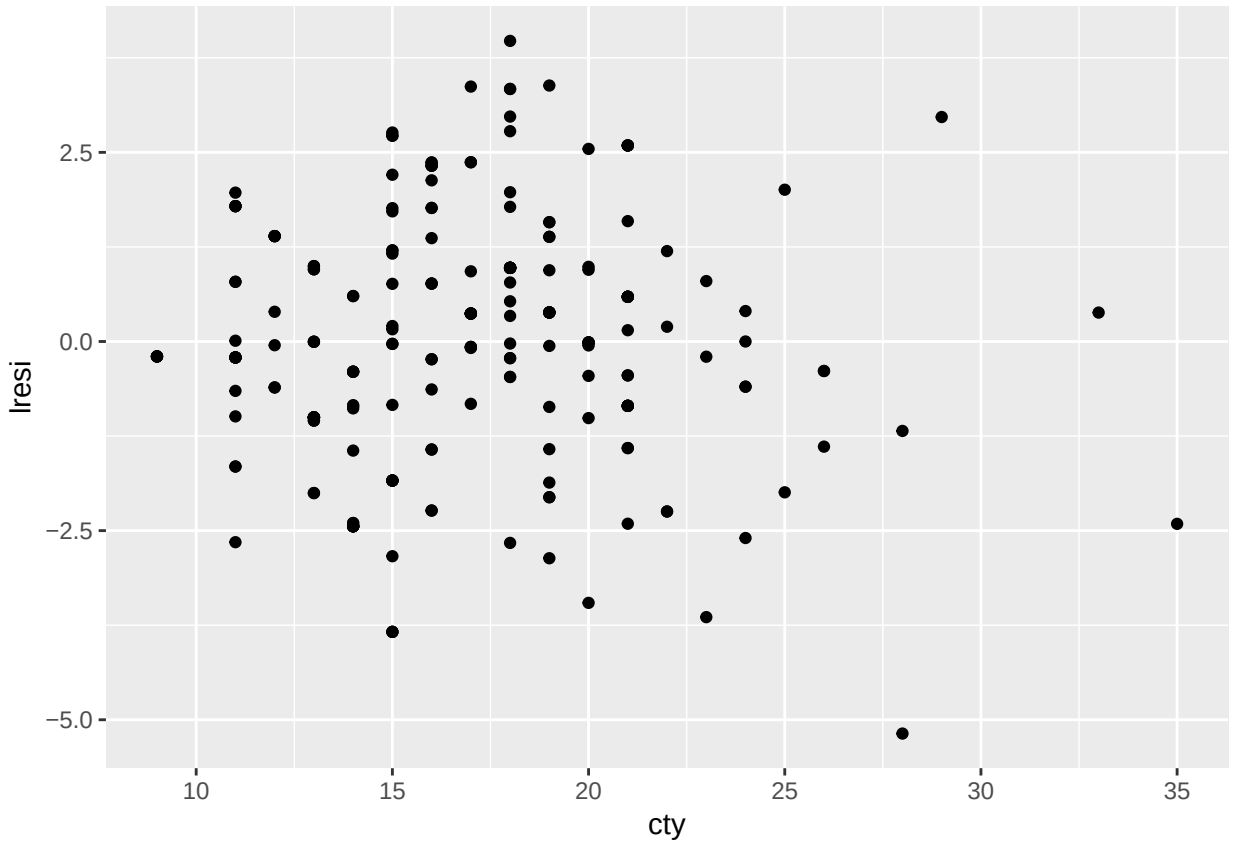
```
rmse(fit_mpg, mpgx)
```

```
## [1] 1.591092
```

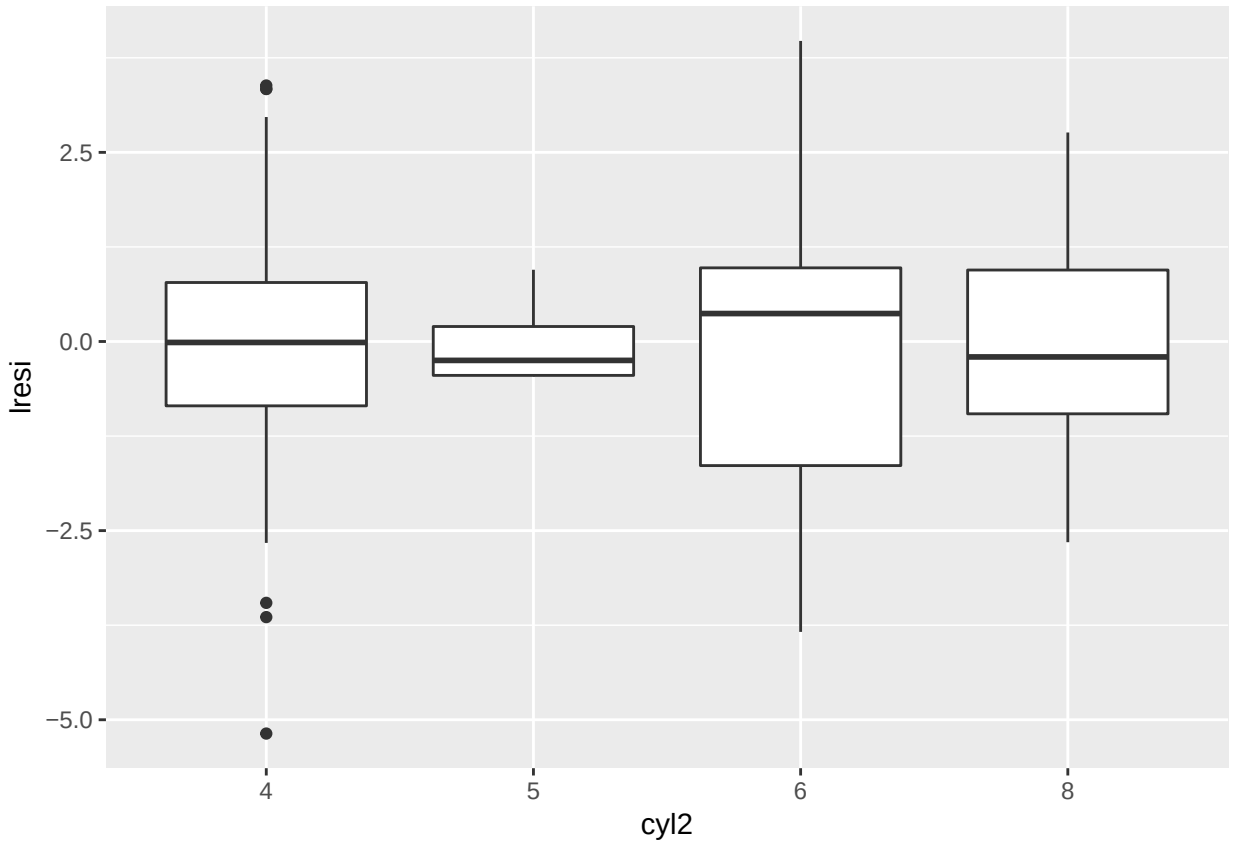
```
mpgx %>% add_residuals(fit_mpg,
    "lresi") %>% ggplot(aes(sample = lresi)) +
    geom_qq()
```



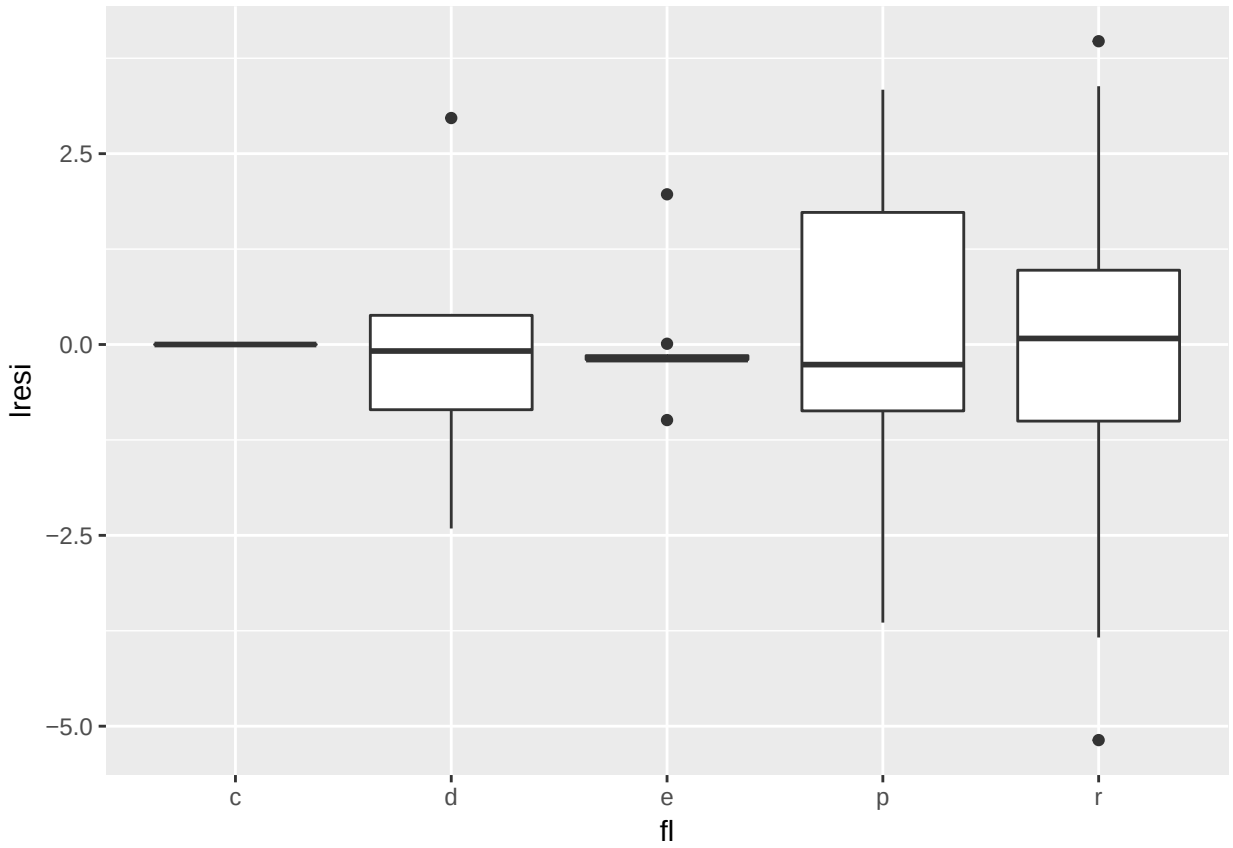

```
mpgx %>% add_residuals(fit_mpg,
    "lresi") %>% ggplot(aes(x = cty,
    y = lresi)) + geom_point()
```




```
mpgx %>% add_residuals(fit_mpg,
    "lresi") %>% ggplot(aes(x = cyl2,
    y = lresi)) + geom_boxplot()
```




```
mpgx %>% add_residuals(fit_mpg,
    "lresi") %>% ggplot(aes(x = fl,
    y = lresi)) + geom_boxplot()
```

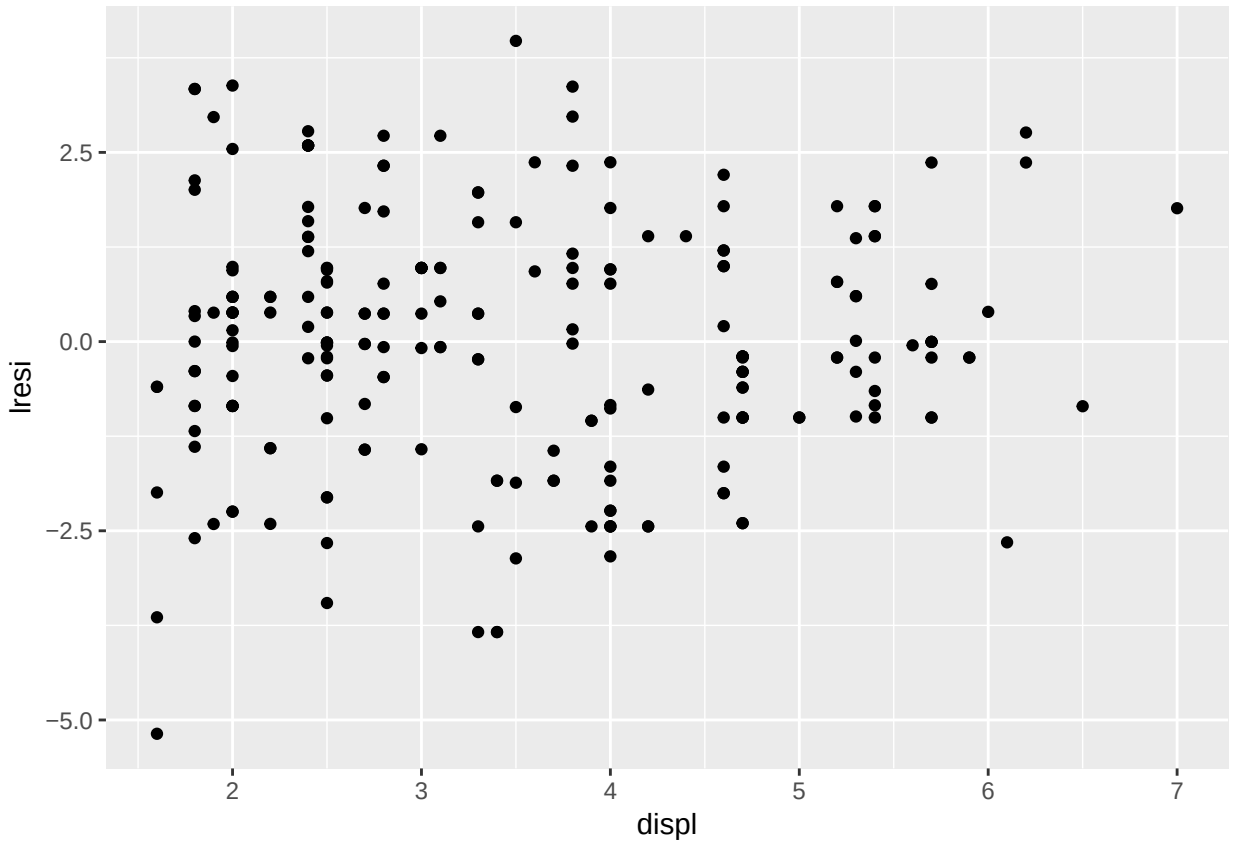


cyl2 seems not random

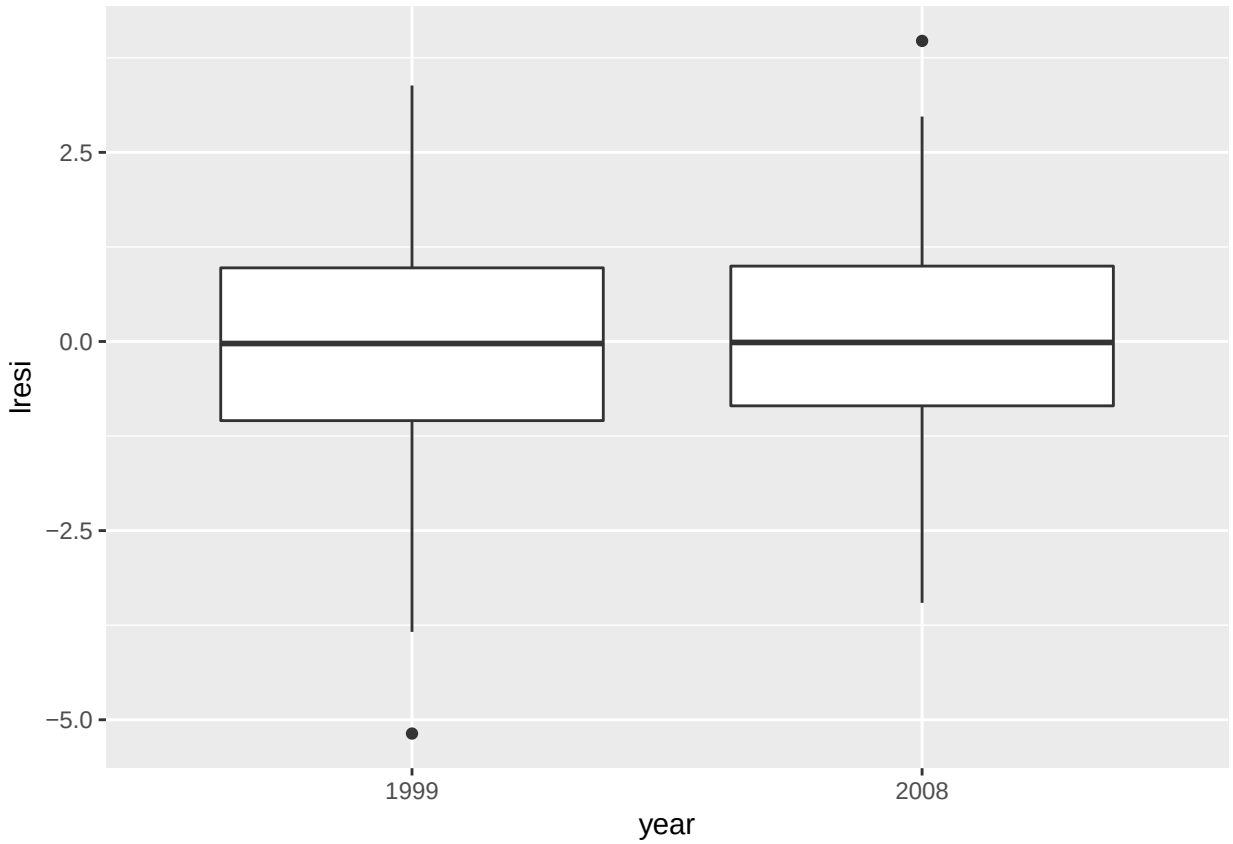
---

prob2.2

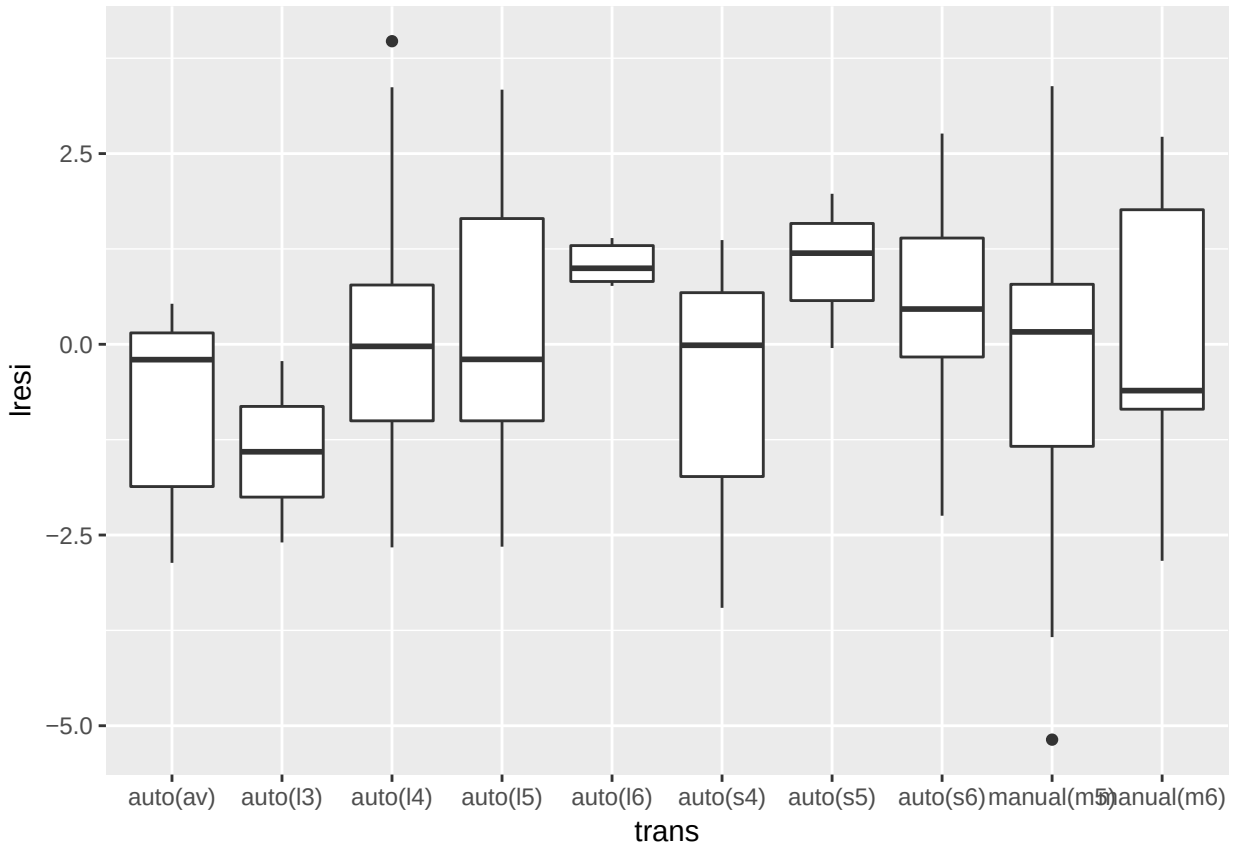
```
resi_mpg <- mpgx %>% add_residuals(fit_mpg,
    "lresi")
ggplot(data = resi_mpg, aes(x = displ,
    y = lresi)) + geom_point()
```



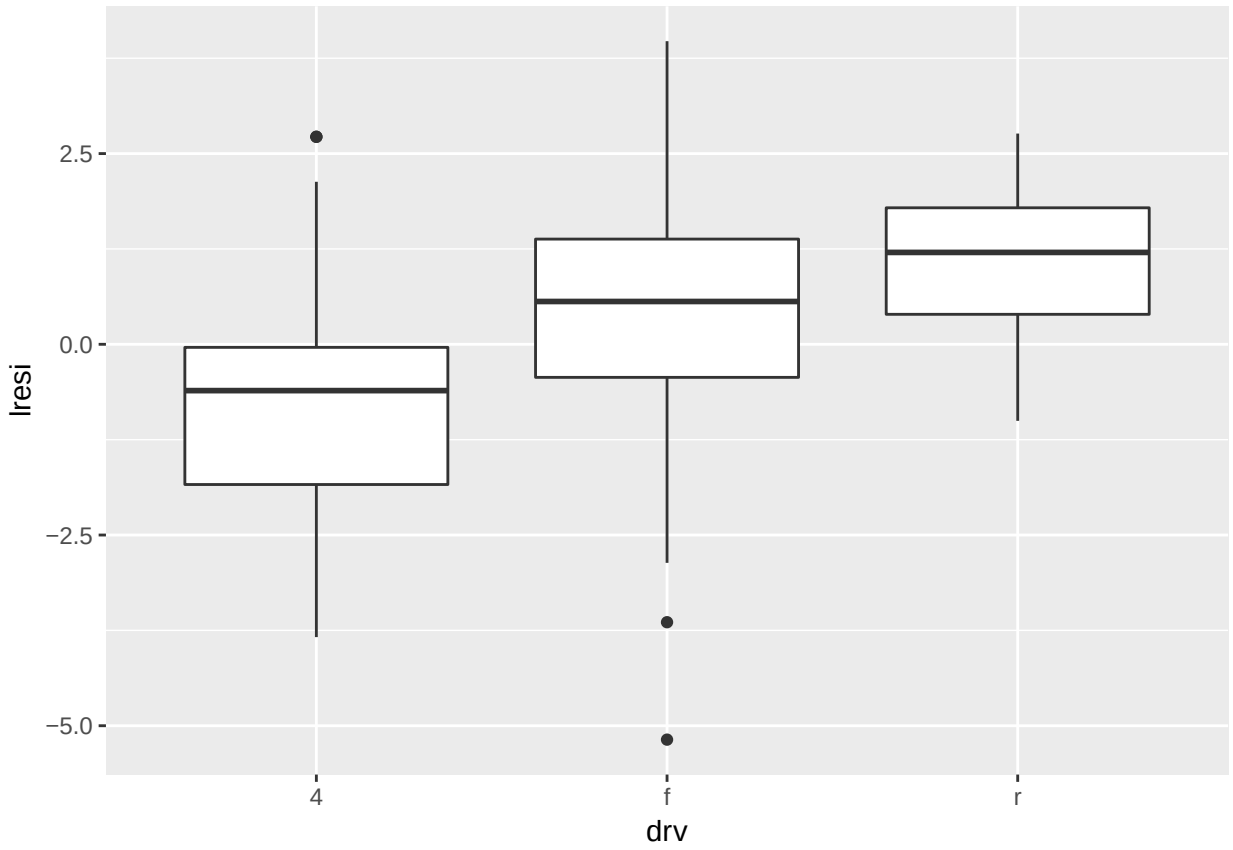

```
ggplot(data = resi_mpg, aes(x = year,
    y = lresi)) + geom_boxplot()
```



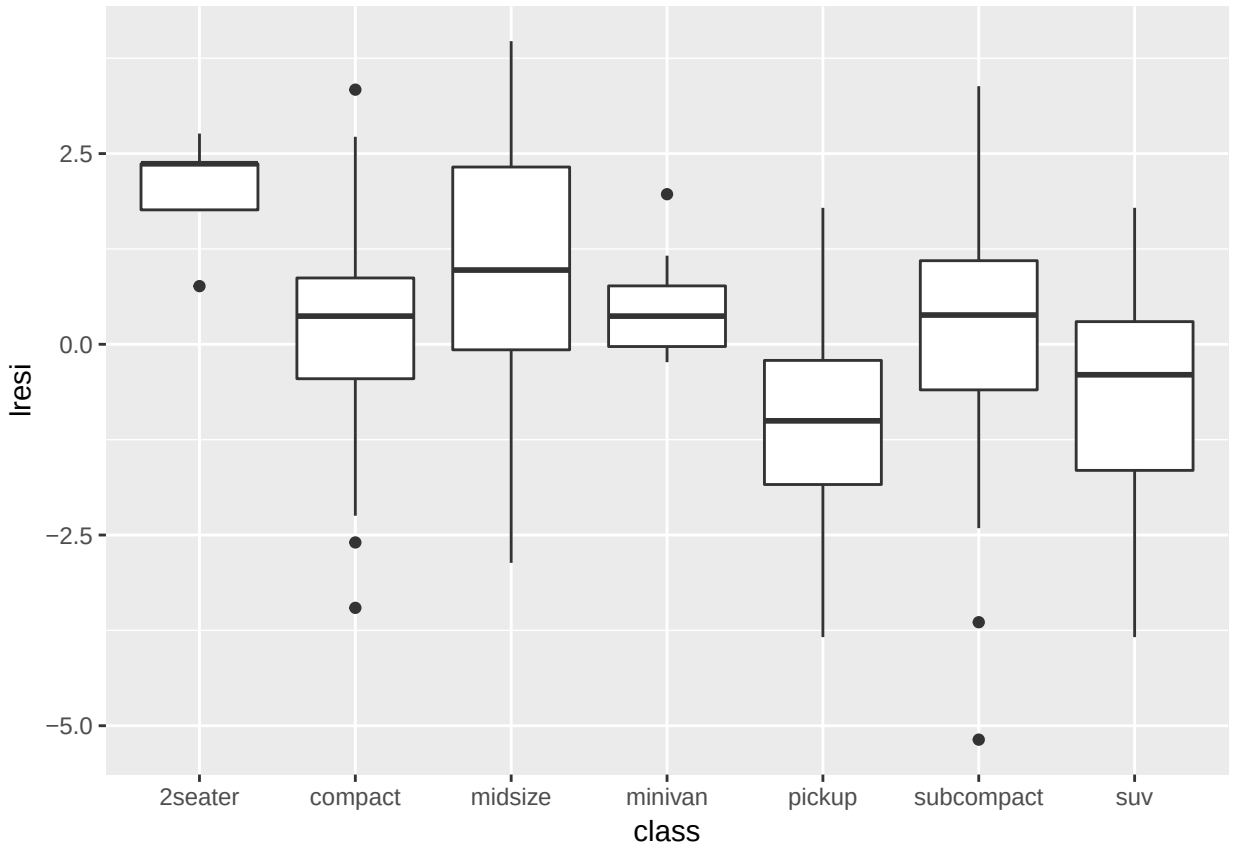

```
ggplot(data = resi_mpg, aes(x = trans,
    y = lresi)) + geom_boxplot()
```




```
ggplot(data = resi_mpg, aes(x = drv,
    y = lresi)) + geom_boxplot()
```




```
ggplot(data = resi_mpg, aes(x = class,
    y = lresi)) + geom_boxplot()
```



trans,drv and class seem not random

---

prob3 add trans, drv, class, remove cyl2

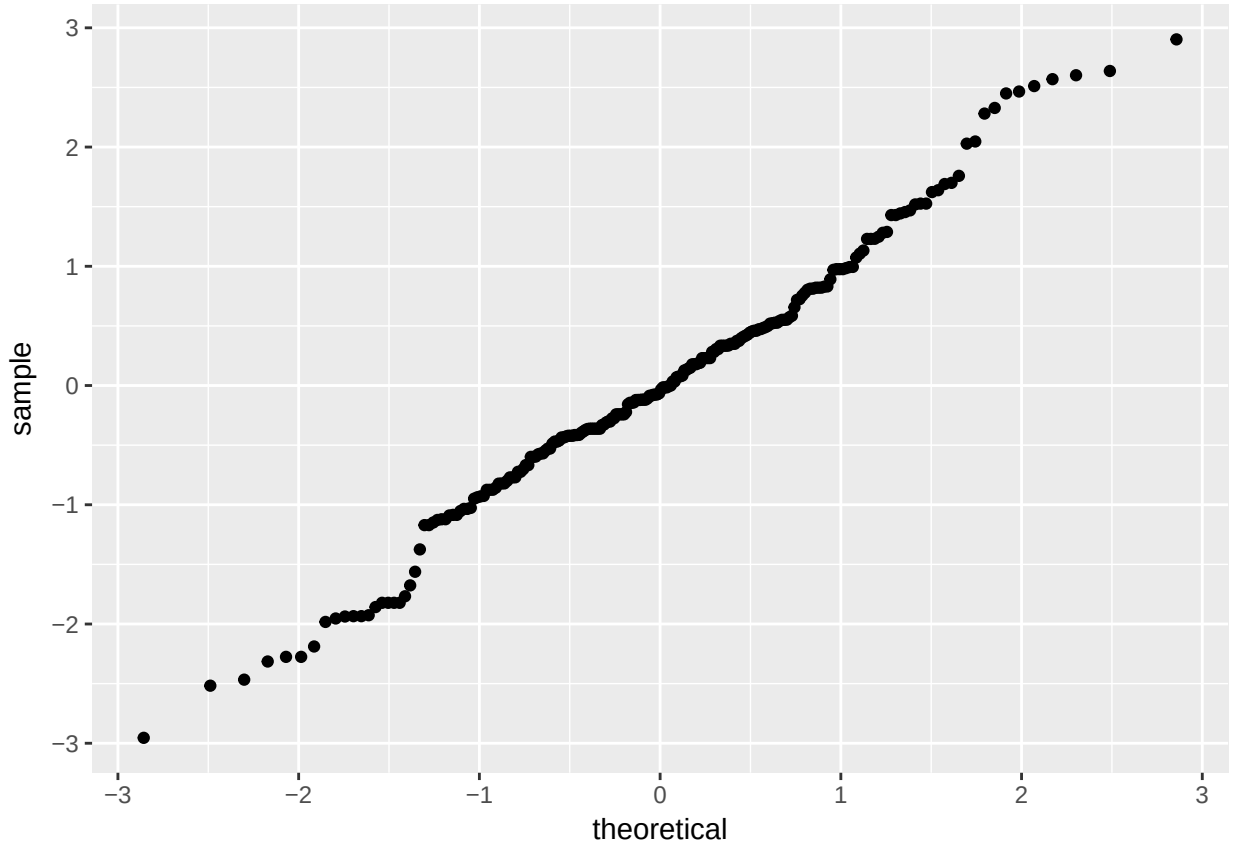
```
fit2_mpg <- lm(hwy ~ cty +
    trans + drv + class +
    fl, data = mpgx)
fit2_mpg
```

```
##
## Call:
## lm(formula = hwy ~ cty + trans + drv + class + fl, data = mpgx)
##
## Coefficients:
##     (Intercept)              cty    transauto(l3)    transauto(l4)
##         9.96720          1.05177         -0.75943          0.76337
##   transauto(l5)    transauto(l6)    transauto(s4)    transauto(s5)
##         1.41502          1.83239         -0.02982          1.88767
##   transauto(s6)  transmanual(m5)  transmanual(m6)             drvf
##         1.17068          0.81649          1.01190          0.86283
##            drvr     classcompact     classmidsize     classminivan
##         1.00880         -0.83819         -0.50935         -2.16943
##     classpickup  classsubcompact         classsuv              fld
##        -3.97994         -1.48760         -3.63080         -2.51024
##              fle              flp              flr
##        -4.74714         -3.36725         -3.65402
```

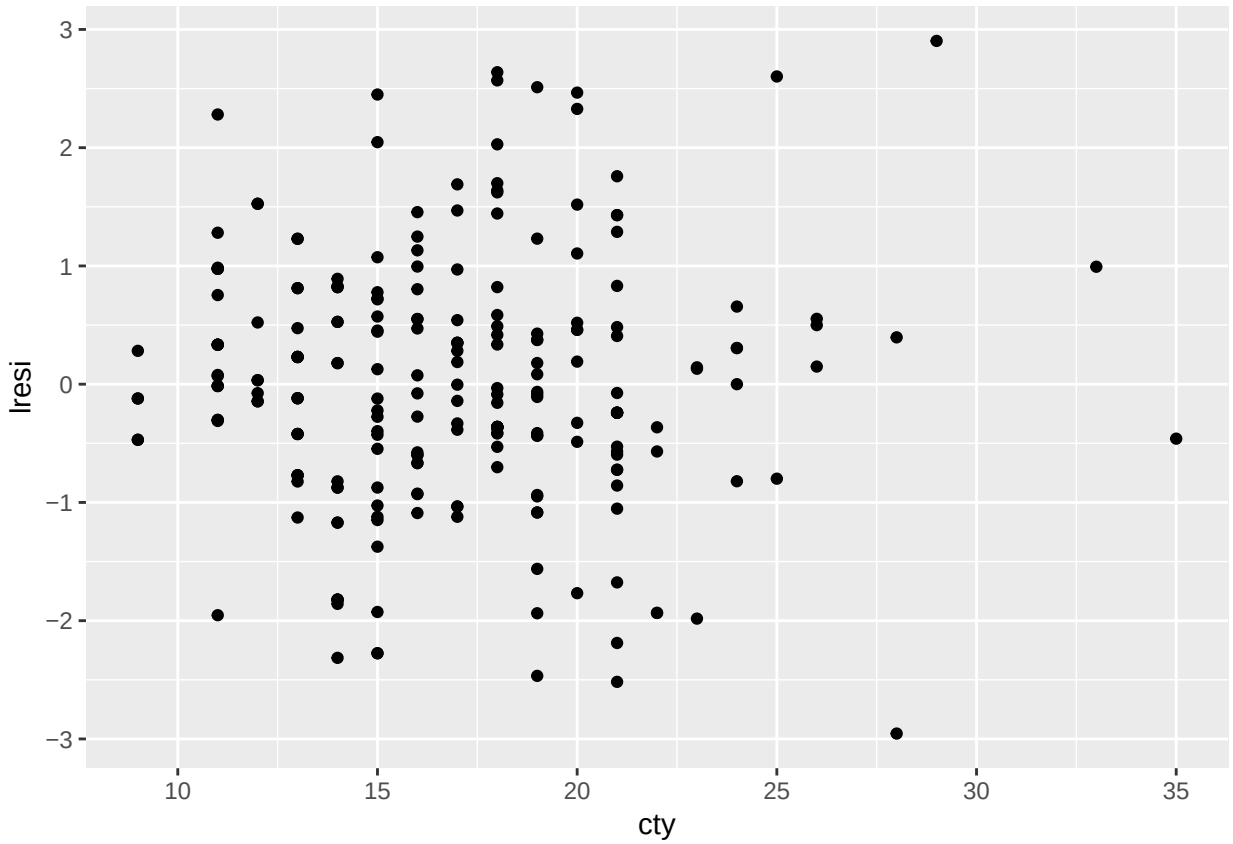
```
rmse(fit2_mpg, mpgx)
```

```
## [1] 1.058914
```

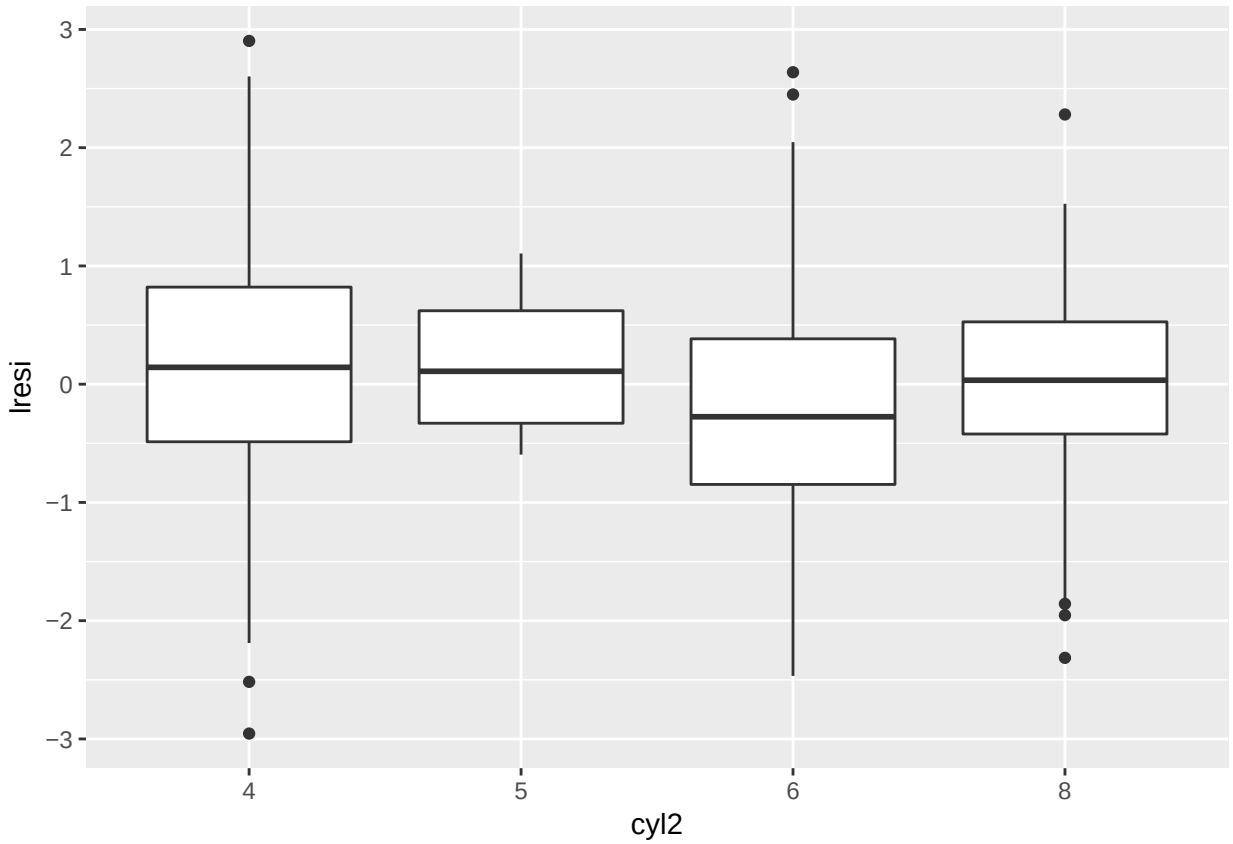
```
mpgx %>% add_residuals(fit2_mpg,
    "lresi") %>% ggplot(aes(sample = lresi)) +
    geom_qq()
```



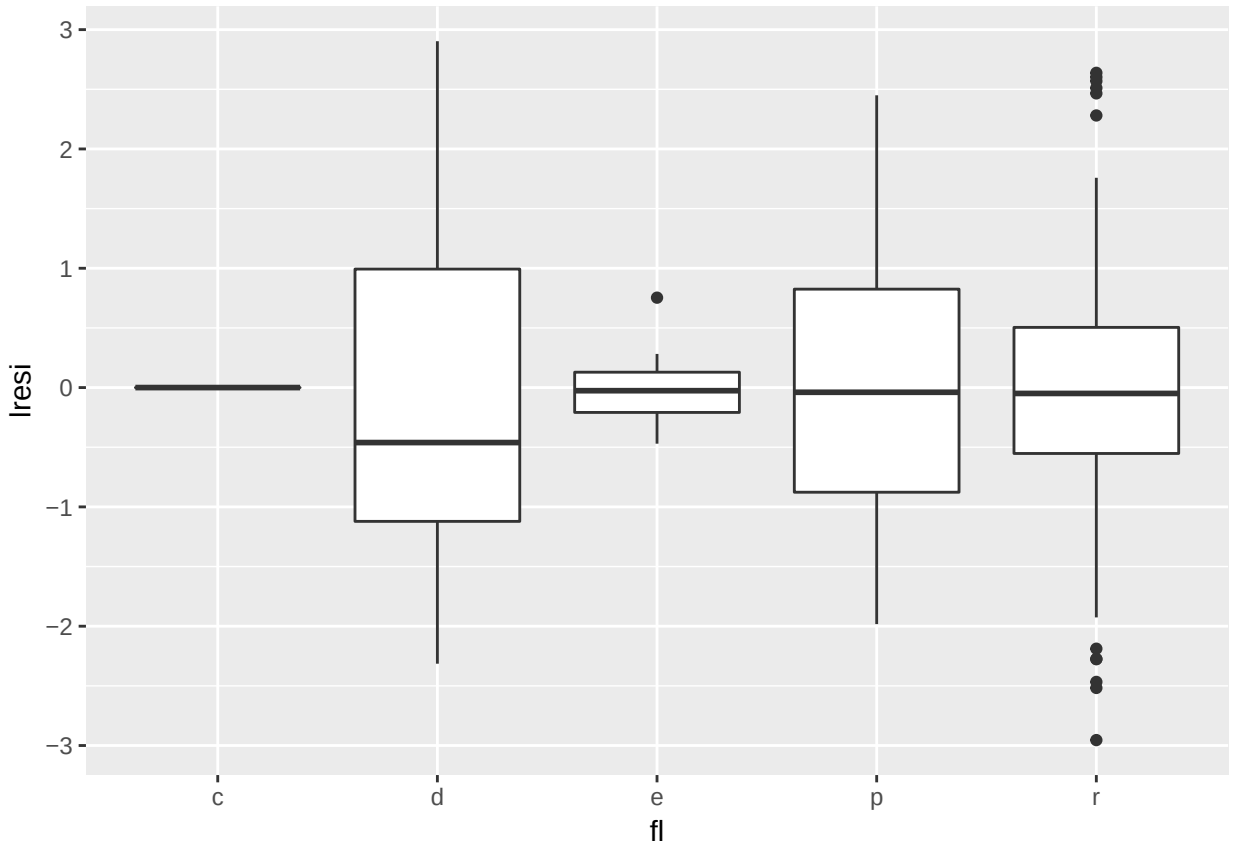

```
mpgx %>% add_residuals(fit2_mpg,
    "lresi") %>% ggplot(aes(x = cty,
    y = lresi)) + geom_point()
```



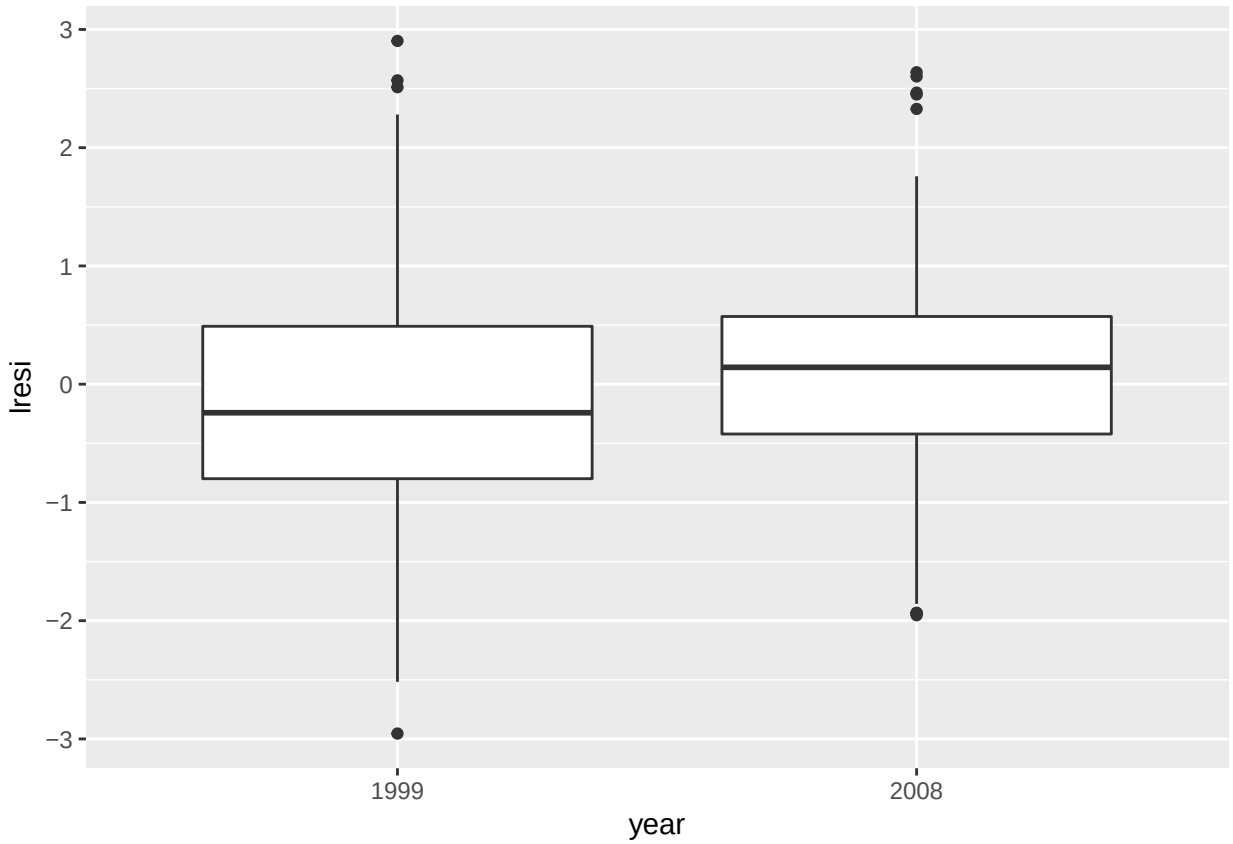

```
mpgx %>% add_residuals(fit2_mpg,
    "lresi") %>% ggplot(aes(x = cyl2,
    y = lresi)) + geom_boxplot()
```



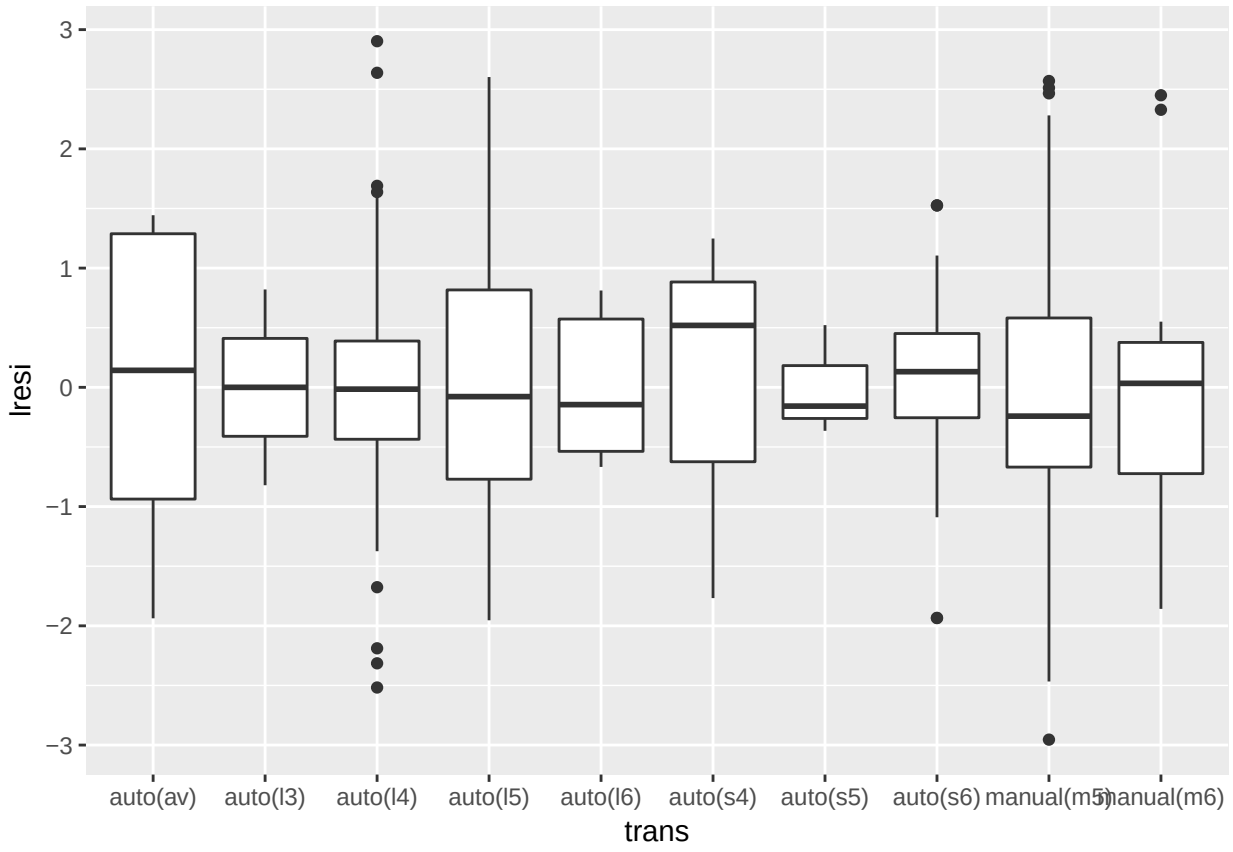

```
mpgx %>% add_residuals(fit2_mpg,
    "lresi") %>% ggplot(aes(x = fl,
    y = lresi)) + geom_boxplot()
```



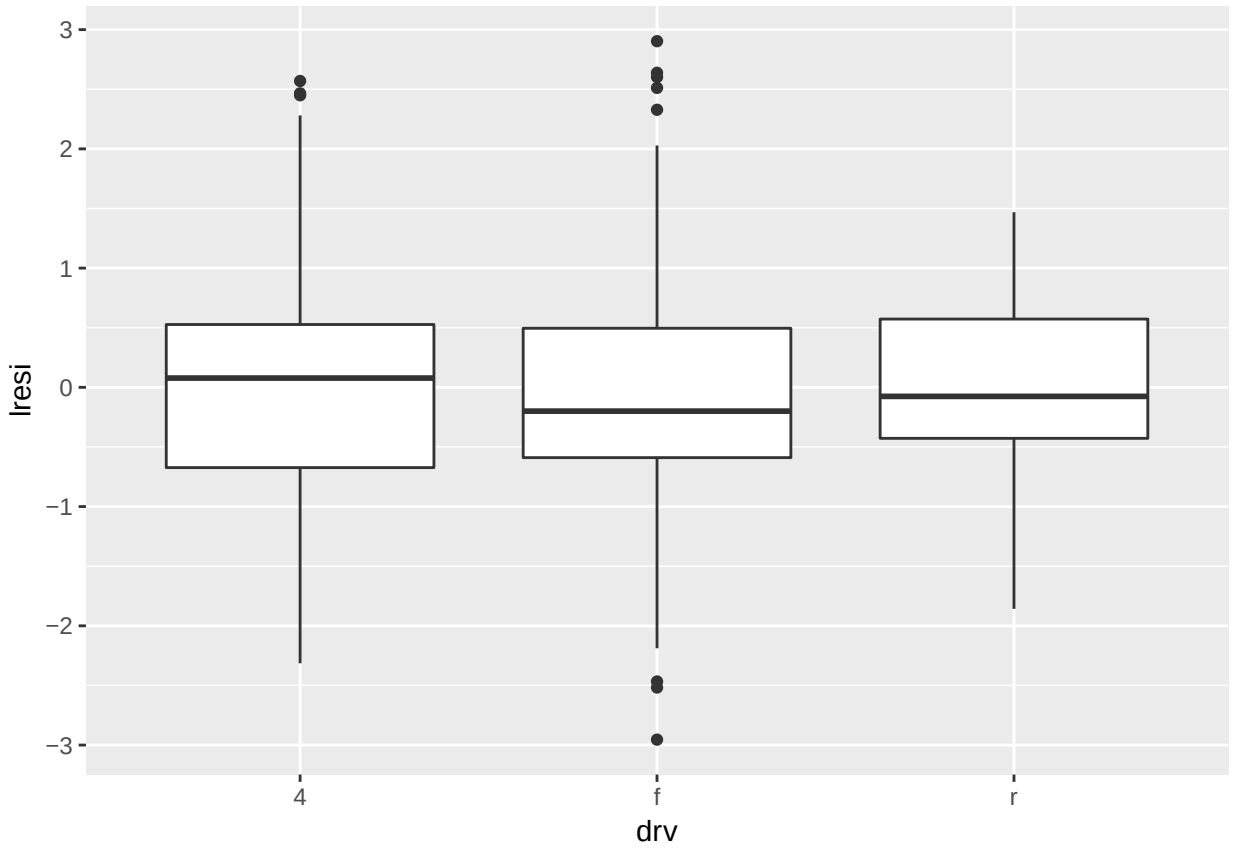

```
resi2_mpg <- mpgx %>% add_residuals(fit2_mpg,
    "lresi")
ggplot(data = resi2_mpg, aes(x = year,
    y = lresi)) + geom_boxplot()
```



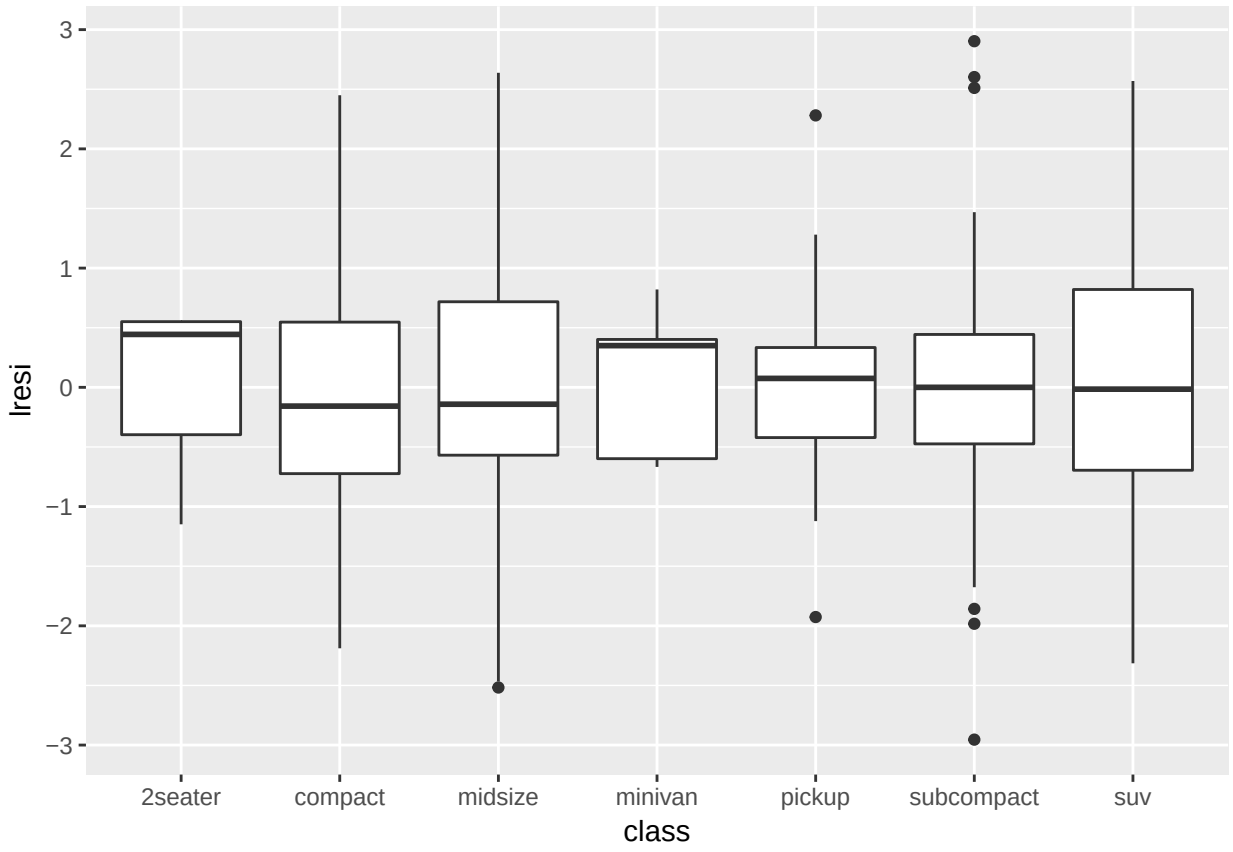

```
ggplot(data = resi2_mpg, aes(x = trans,
    y = lresi)) + geom_boxplot()
```




```
ggplot(data = resi2_mpg, aes(x = drv,
    y = lresi)) + geom_boxplot()
```




```
ggplot(data = resi2_mpg, aes(x = class,
    y = lresi)) + geom_boxplot()
```



---

prob4&5

Note:since there is a problem when new values in test are not seen in train, fl is not used in cross-validation. Teacher said it's OK to do so.

```
crossvalid <- function(model_formula,
    dataset, num_fold) {
    set.seed(2)
    crossv <- crossv_kfold(dataset,
        num_fold)  # dataset should be data frame?
    data_fit <- mutate(crossv,
        fit = map(train, ~lm(model_formula,
            data = .)))
    errorx <- transmute(data_fit,
        train_error = map2_dbl(fit,
            train, ~rmse(.x,
                .y)), test_error = map2_dbl(fit,
            test, ~rmse(.x,
                .y)))
    return(errorx %>% summarize(mean(train_error),
        mean(test_error)))
}
crossvalid(hwy ~ cty + cyl2,
    mpgx, 10)
```

```
## # A tibble: 1 x 2
##   `mean(train_error)` `mean(test_error)`
```

```
##                 <dbl>              <dbl>
## 1            1.730239           1.746031
```

```
crossvalid(hwy ~ cty + trans +
    drv + class, mpgx, 10)
```

```
## # A tibble: 1 x 2
##   `mean(train_error)` `mean(test_error)`
##                 <dbl>              <dbl>
## 1            1.102738           1.227221
```

```
crossvalid(hwy ~ cty + drv +
    class, mpgx, 10)
```

```
## # A tibble: 1 x 2
##   `mean(train_error)` `mean(test_error)`
##                 <dbl>              <dbl>
## 1            1.175645           1.210007
```

```
crossvalid(hwy ~ cty + drv +
    class + cyl2, mpgx, 10)
```

```
## # A tibble: 1 x 2
##   `mean(train_error)` `mean(test_error)`
##                 <dbl>              <dbl>
## 1            1.172912           1.217239
```

So the model(hwy~cty+trans+drv+class) in Prob3 is better(not considering fl). But the model hwy~cty+drv+class is the best.

---

prob6

```
library(xml2)
imzml <- read_xml("/Users/yzh/Desktop/R data/HW3/Example_Continuous.imzML")

insert_ref_groups <- function(x) {
    ref_groups <- xml_root(x) %>%
        xml_child("d1:referenceableParamGroupList") %>%
        xml_children()
    ref <- xml_child(x, "d1:referenceableParamGroupRef")
    name <- xml_attr(ref,
        "ref")
    ref_groups_exist <- xml_attr(ref_groups,
        "id") %in% name
    if (any(ref_groups_exist))
        group <- ref_groups[[which(ref_groups_exist)]]
    for (g in xml_children(group)) xml_add_child(x,
        g)
    xml_remove(ref)
    x
}

xml_find_by_attribute <- function(x,
    attr, value) {
    match <- xml_attr(x, attr) ==
        value
```

```
    if (isTRUE(any(match))) {
        x[[which(match)]]
    } else {
        NULL
    }
}

get_spectrum_data <- function(x,
    i) {
    spectrum <- x %>% xml_child("d1:run") %>%
        xml_child("d1:spectrumList") %>%
        xml_child(i)
    spectrum <- insert_ref_groups(spectrum)
    scan <- spectrum %>% xml_child("d1:scanList") %>%
        xml_child("d1:scan")
    scan <- insert_ref_groups(scan)
    data <- spectrum %>% xml_child("d1:binaryDataArrayList") %>%
        xml_children()
    for (d in data) insert_ref_groups(d)
    data <- lapply(data, xml_children)
    for (i in seq_along(data)) {
        if (!is.null(xml_find_by_attribute(data[[i]],
            "name", "m/z array")))
            names(data)[i] <- "mz"
        if (!is.null(xml_find_by_attribute(data[[i]],
            "name", "intensity array")))
            names(data)[i] <- "intensity"
    }
    data$coord <- xml_children(scan)
    data[c("mz", "intensity",
        "coord")]
}

get_spectra_n <- function(x) {
    x %>% xml_child("d1:run") %>%
        xml_child("d1:spectrumList") %>%
        xml_attr("count") %>%
        as.numeric()
}

get_spectra <- function(x) {
    n <- get_spectra_n(x)
    lapply(1:n, function(i) get_spectrum_data(x,
        i))
}

x <- imzml
get_spectra_n(x)
```

```
## [1] 9
```

```
spectra_info <- get_spectra(x)

coord_x <- spectra_info %>%
```

```
    map_dbl(~xml_find_by_attribute(.$coord,
        "name", "position x") %>%
        xml_attr("value") %>%
        as.numeric())
coord_x
```

```
## [1] 1 2 3 1 2 3 1 2 3
```

```
coord_y <- spectra_info %>%
    map_dbl(~xml_find_by_attribute(.$coord,
        "name", "position y") %>%
        xml_attr("value") %>%
        as.numeric())
coord_y
```

```
## [1] 1 1 1 2 2 2 3 3 3
```

```
mz_length <- spectra_info[[1]]$mz %>%
    xml_find_by_attribute("name",
        "external array length") %>%
    xml_attr("value") %>%
    as.numeric()
mz_length
```

```
## [1] 8399
```

```
mz_offset <- spectra_info[[1]]$mz %>%
    xml_find_by_attribute("name",
        "external offset") %>%
    xml_attr("value") %>%
    as.numeric()
mz_offset
```

```
## [1] 16
```

```
intensity_length <- spectra_info %>%
    map_dbl(~xml_find_by_attribute(.$intensity,
        "name", "external array length") %>%
        xml_attr("value") %>%
        as.numeric())
intensity_length
```

```
## [1] 8399 8399 8399 8399 8399 8399 8399 8399 8399
```

```
intensity_offset <- spectra_info %>%
    map_dbl(~xml_find_by_attribute(.$intensity,
        "name", "external offset") %>%
        xml_attr("value") %>%
        as.numeric())
intensity_offset
```

```
## [1]  33612  67208 100804 134400 167996 201592 235188 268784 302380
```

---

prob7

```
get_mz_intensity_arrays <- function() {
    filename <- "/Users/yzh/Desktop/R data/HW3/Example_Continuous.ibd"
```

```
    intensity <- map2(intensity_offset,
        intensity_length,
        function(offset, length) {
            f <- file(filename,
                "rb")
            seek(f, offset)
            iout <- readBin(f,
                "double",
                n = length,
                size = 4)
            close(f)
            iout
        })
    f <- file(filename, "rb")
    seek(f, mz_offset)
    mz <- readBin(f, "double",
        n = mz_length, size = 4)
    close(f)
    return(list(mz, intensity))
}
mz <- get_mz_intensity_arrays()[[1]]
intensity <- get_mz_intensity_arrays()[[2]]
```

---

prob8

```
construct <- function(mz,
    intensity, coord_x, coord_y) {
    structure(list(mz = mz,
        intensity = simplify2array(intensity),
        coord = tibble(x = coord_x,
            y = coord_y)),
        class = "msi")
}
msi <- construct(mz, intensity,
    coord_x, coord_y)
```

---

prob9 access_method

```
access <- function(object) UseMethod("access")
access.msi <- function(object) return(object)
```

---

prob9 plot_method

```
msi <- access(msi)

plot_msi <- function(a, b) UseMethod("plot_msi")
plot_msi.msi <- function(x,
    mz) {
    idx <- which.min(abs(mz -
        x$mz))
    idf <- x$coord
```

```
    idf$intensity <- x$intensity[idx,
        ]
    ggplot(idf) + geom_tile(aes(x = x,
        y = y, fill = intensity)) +
        scale_y_reverse()
}
plot_msi(msi, 151.9)
```