STATS 3001 / STATS 4104 / STATS 7054
Statistical Modelling III
Practical 2 - Factors
Solutions

Week 2

GOAL

This practical is intended to illustrate some of the properties of linear models involving factors and their implementation in R.

DATA

The dataset `loan.xlsx` - get it from MyUni - has the following variables.

Var	Description
`loan_amnt`	the amount of the loan in dollars
`term`	the term of the loan in months
`home_ownership`	home ownership status (rent/own/mortgage)
`annual_inc`	the annual income of the applicant in dollars

STEPS

- Read in the data

```
loan <- readxl::read_excel(here::here("data", "loan.xlsx"))
loan
```

```
## # A tibble: 30 x 4
##    loan_amnt term      home_ownership annual_inc
##        <dbl> <chr>     <chr>               <dbl>
##  1      8000 36 months OWN                 32800
##  2     11200 36 months OWN                 46000
##  3      1000 36 months OWN                 77367
##  4      7000 36 months OWN                 56004
##  5      1400 36 months OWN                 11000
##  6      4800 60 months OWN                 40000
##  7     12000 60 months OWN                 18000
##  8     20000 60 months OWN                120000
##  9     21600 60 months OWN                 60000
## 10      4000 60 months OWN                 42000
## # ... with 20 more rows
```

- Fit the model,

```
loan_amnt ~ home_ownership
```

```
loan_lm1 <- lm(loan_amnt ~ home_ownership, data = loan)
```

- What is the reference category for `home_ownership`?

```
loan %>% count(home_ownership)
```

```
## # A tibble: 3 x 2
##   home_ownership     n
##   <chr>          <int>
## 1 MORTGAGE          10
## 2 OWN               10
## 3 RENT              10
```

```
model.matrix(loan_lm1)[1:5, ]
```

```
##   (Intercept) home_ownershipOWN home_ownershipRENT
## 1           1                 1                  0
## 2           1                 1                  0
## 3           1                 1                  0
## 4           1                 1                  0
## 5           1                 1                  0
```

The reference category is `MORTGAGE`.

- Calculate the group means for each level of `home_ownership`. Show how these can be obtained from the `lm()` output.

```
grp_means <-
  loan %>%
  group_by(home_ownership) %>%
  summarise(mean(loan_amnt), .groups = "keep")
grp_means
```

```
## # A tibble: 3 x 2
## # Groups:   home_ownership [3]
##   home_ownership ‘mean(loan_amnt)‘
##   <chr>                     <dbl>
## 1 MORTGAGE                  17185
## 2 OWN                        9100
## 3 RENT                      14280
```

```
loan_lm1 %>%
  tidy()
```

```
## # A tibble: 3 x 5
##   term               estimate std.error statistic     p.value
##   <chr>                 <dbl>     <dbl>     <dbl>       <dbl>
## 1 (Intercept)           17185     2456.     7.00  0.000000161
## 2 home_ownershipOWN     -8085     3474.    -2.33  0.0277
## 3 home_ownershipRENT    -2905     3474.    -0.836 0.410
```

So group mean for `MORTGAGE` is the intercept. The group mean for `OWN` is the intercept minus 8085, and finally the group mean of `RENT` is the intercept minus 2905.

- Redo the linear modelling using the zero sum constraint.

Redo model with new constraint

```
loan_lm2 <- lm(loan_amnt ~ home_ownership,
               data=loan,
               contrasts = list(home_ownership = "contr.sum")
)
model.matrix(loan_lm2)[1:5,]
```

```
##   (Intercept) home_ownership1 home_ownership2
## 1           1               0               1
## 2           1               0               1
## 3           1               0               1
## 4           1               0               1
## 5           1               0               1
```

- Calculate the overall mean `loan_amnt`. How can you get this, and the group means from the new `lm()` output?

```
mean(loan$loan_amnt)
```

```
## [1] 13521.67
```

```
grp_means
```

```
## # A tibble: 3 x 2
## # Groups:   home_ownership [3]
##   home_ownership ‘mean(loan_amnt)‘
##   <chr>                     <dbl>
## 1 MORTGAGE                  17185
## 2 OWN                        9100
## 3 RENT                      14280
```

```
loan_lm2 %>% tidy()
```

```
## # A tibble: 3 x 5
##   term            estimate std.error statistic  p.value
##   <chr>              <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)       13522.     1418.      9.53 3.91e-10
## 2 home_ownership1    3663.     2006.      1.83 7.89e- 2
## 3 home_ownership2   -4422.     2006.     -2.20 3.62e- 2
```

So the intercept is the overall mean.

So group mean for `MORTGAGE` is the intercept plus 3663.333. The group mean for `OWN` is the intercept minus 4421.667, and finally the group mean of `RENT` is the intercept minus 3663.333 and plus 4421.667.

- Fit the models

```
loan_amnt ~ home_ownership + annual_inc
loan_amnt ~ home_ownership * annual_inc
```

```
loan_lm3 <- lm(loan_amnt ~ home_ownership + annual_inc,data=loan)
loan_lm4 <- lm(loan_amnt ~ home_ownership * annual_inc,data=loan)
loan_lm4 <- lm(loan_amnt ~ home_ownership + annual_inc + home_ownership:annual_inc,data=loan)
```

- For each model, give the estimated regression line for each of the three groups.

```
loan_lm3 %>% tidy()
```

```
## # A tibble: 4 x 5
##   term                estimate std.error statistic p.value
##   <chr>                  <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)         9699.    3513.          2.76   0.0104
## 2 home_ownershipOWN  -5734.    3235.         -1.77   0.0880
## 3 home_ownershipRENT -2343.    3126.         -0.749  0.460
## 4 annual_inc             0.102    0.0373     2.74   0.0110
```

MORTGAGE

loan_amnt = 9698.62 + 0.1021 annual_inc

OWN

loan_amnt = (9698.62 - 5734.09) + 0.1021 annual_inc

RENT

loan_amnt = (9698.62 - 2342.52) + 0.1021 annual_inc

```
loan_lm4 %>% tidy()
```

```
## # A tibble: 6 x 5
##   term                          estimate std.error statistic p.value
##   <chr>                            <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)                  11514.     5015.        2.30   0.0307
## 2 home_ownershipOWN            -7875.     6731.       -1.17   0.254
## 3 home_ownershipRENT           -5780.     7074.       -0.817  0.422
## 4 annual_inc                       0.0773    0.0609    1.27   0.216
## 5 home_ownershipOWN:annual_inc     0.0312    0.0981    0.318  0.753
## 6 home_ownershipRENT:annual_inc    0.0487    0.0894    0.544  0.591
```

MORTGAGE

loan_amnt = 11514.42 + 0.0773 annual_inc

OWN

loan_amnt = (11514.42 - 7875.047) + (0.0773 + 0.0312) annual_inc

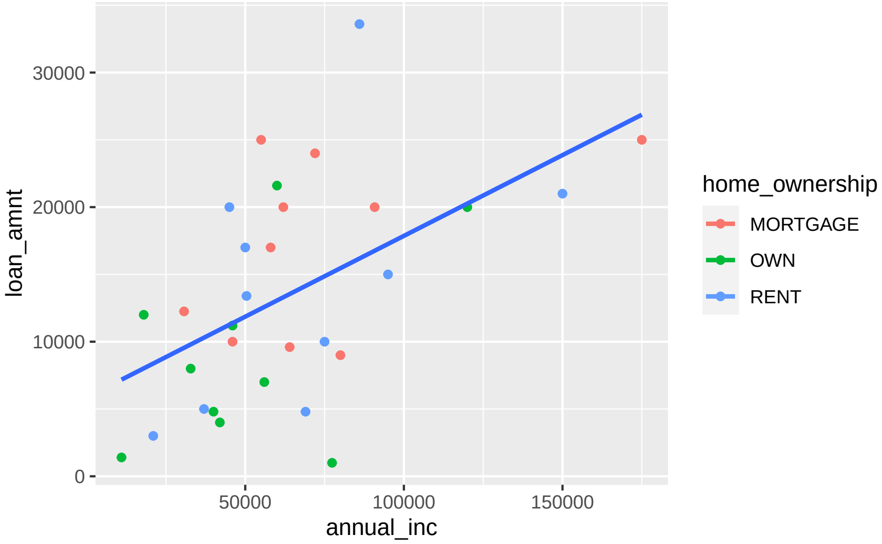
RENT

loan_amnt = (11514.42 -5779.526) + (0.0773 + 0.0487) annual_inc

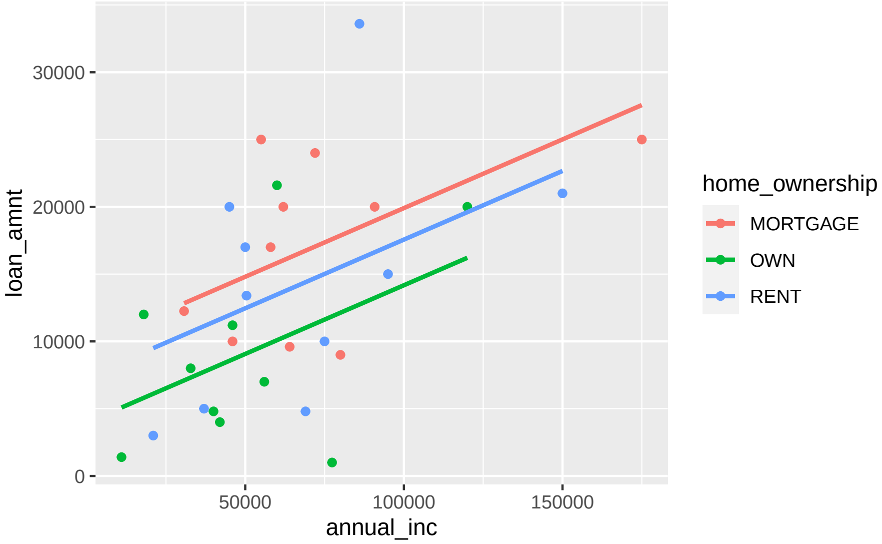
Plots

```
## Identical regression ----
loan %>%
  ggplot(aes(annual_inc, loan_amnt, col = home_ownership)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE, aes(group = 1))
```

```
## `geom_smooth()` using formula 'y ~ x'
```

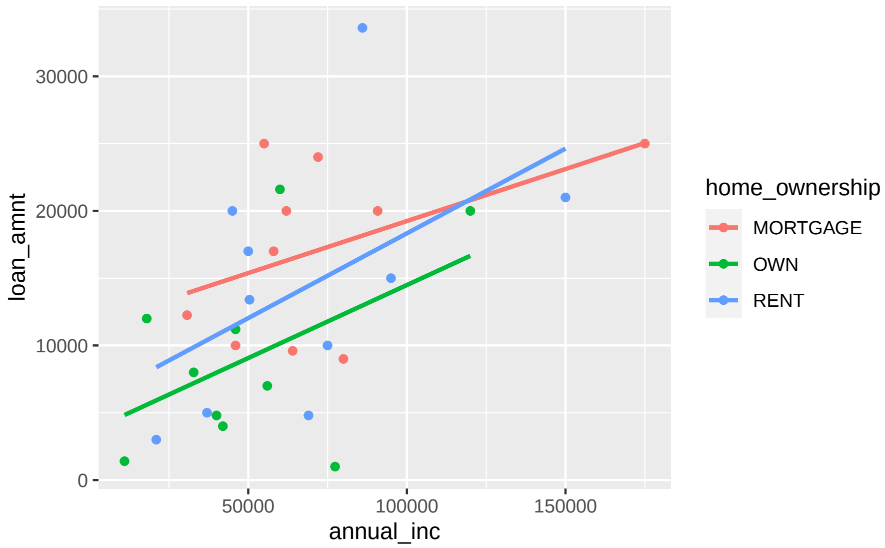


```
## Parallel regression ----
broom::augment(loan_lm3) %>%
  ggplot(aes(annual_inc, loan_amnt, col = home_ownership)) +
  geom_point() +
  geom_line(aes(y = .fitted), size = 1)
```



```
## Separate regression ----
loan %>%
  ggplot(aes(annual_inc, loan_amnt, col = home_ownership)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE)
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



Which model?

```
summary(loan_lm4)
```

```
##
## Call:
## lm(formula = loan_amnt ~ home_ownership + annual_inc + home_ownership:annual_inc,
##     data = loan)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11035.6  -4852.3   -841.2   3603.8  17032.6
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    1.151e+04  5.015e+03   2.296   0.0307 *
## home_ownershipOWN             -7.875e+03  6.731e+03  -1.170   0.2535
## home_ownershipRENT            -5.780e+03  7.074e+03  -0.817   0.4220
## annual_inc                     7.731e-02  6.088e-02   1.270   0.2163
## home_ownershipOWN:annual_inc   3.122e-02  9.806e-02   0.318   0.7530
## home_ownershipRENT:annual_inc  4.865e-02  8.936e-02   0.544   0.5912
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
```

```
## Residual standard error: 7213 on 24 degrees of freedom
## Multiple R-squared:  0.3644, Adjusted R-squared:  0.232 
## F-statistic: 2.752 on 5 and 24 DF,  p-value: 0.04206
```

```
anova(loan_lm4)
```

```
## Analysis of Variance Table
## 
## Response: loan_amnt
##                           Df     Sum Sq   Mean Sq F value  Pr(>F)  
## home_ownership             2  335462167 167731083  3.2237 0.05754 .
## annual_inc                 1  364596881 364596881  7.0073 0.01411 *
## home_ownership:annual_inc  2   15913787   7956894  0.1529 0.85902  
## Residuals                 24 1248745582  52031066                  
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
anova(loan_lm4, loan_lm3)
```

```
## Analysis of Variance Table
## 
## Model 1: loan_amnt ~ home_ownership + annual_inc + home_ownership:annual_inc
## Model 2: loan_amnt ~ home_ownership + annual_inc
##   Res.Df        RSS Df Sum of Sq      F Pr(>F)
## 1     24 1248745582                           
## 2     26 1264659369 -2 -15913787 0.1529  0.859
```

```
anova(loan_lm3, loan_lm2)
```

```
## Analysis of Variance Table
## 
## Model 1: loan_amnt ~ home_ownership + annual_inc
## Model 2: loan_amnt ~ home_ownership
##   Res.Df        RSS Df  Sum of Sq      F  Pr(>F)  
## 1     26 1264659369                               
## 2     27 1629256250 -1 -364596881 7.4957 0.01101 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```