Parameter Interpretation - IFLS data

Sarah Teichman

03/05/20

```
fit <- fit_ifls
dat <- deflate_y_dat
post <- as.data.frame(fit)
fit_summ <- as.data.frame(summary(fit)$summary)
```

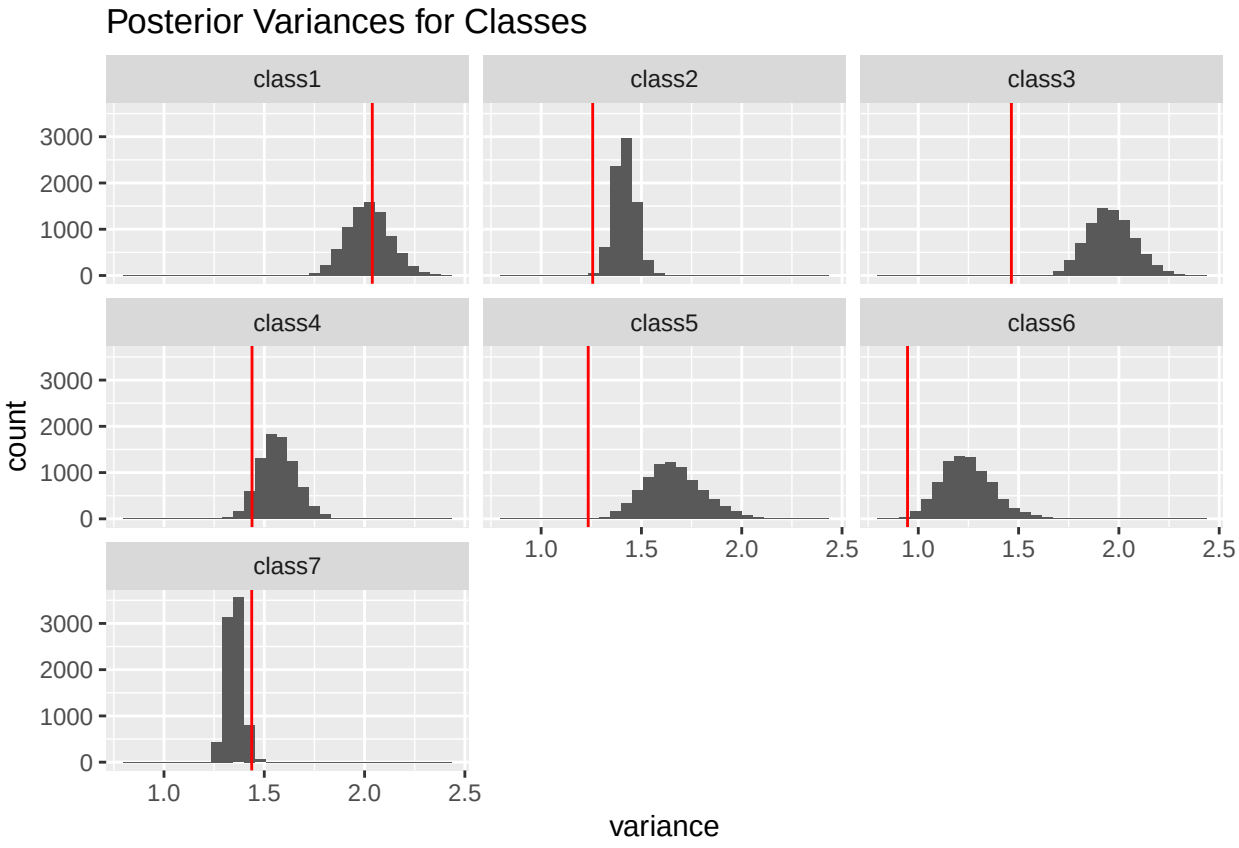
```
K <- 7
Ti <- 3
N <- 1973
```

Posterior histograms for σ_k^2

Plotting posterior histograms from σ_k^2 parameters.

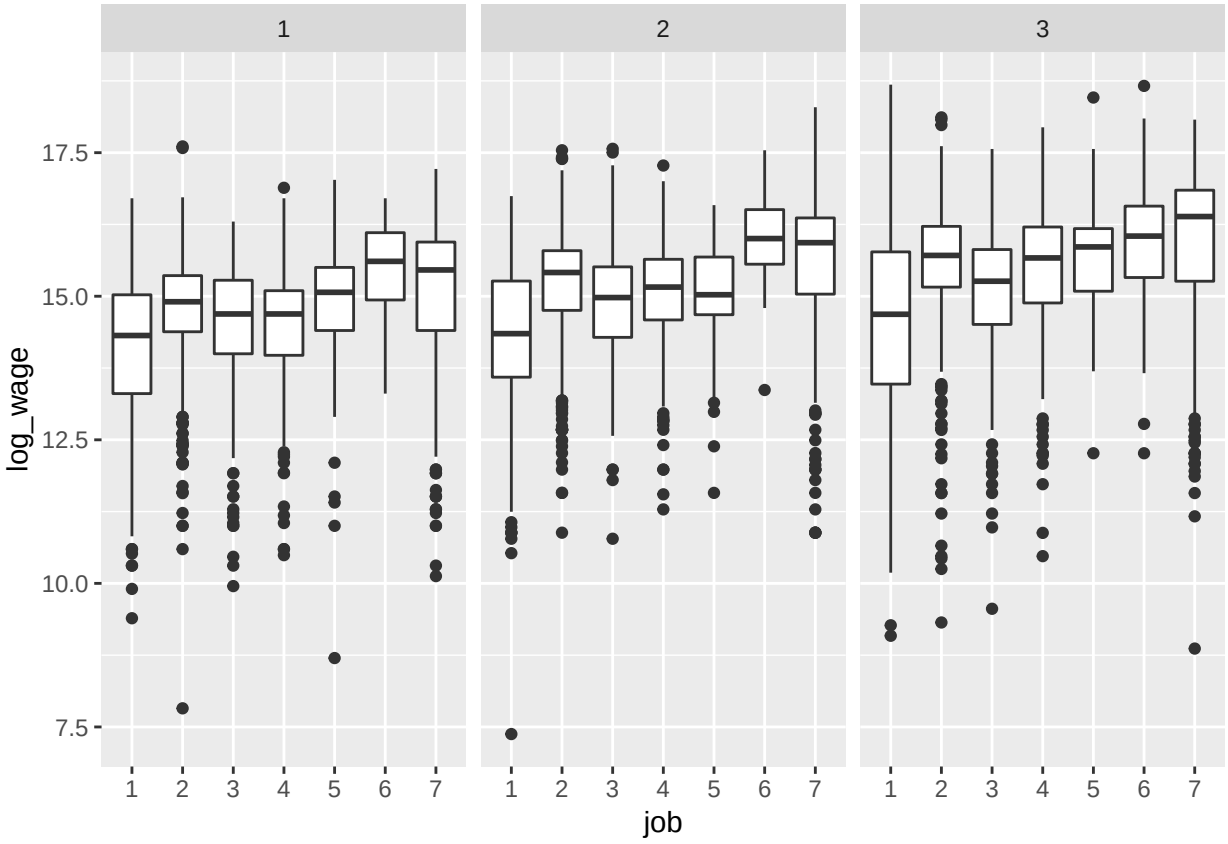
Computing empirical wage variances across firm classes.

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

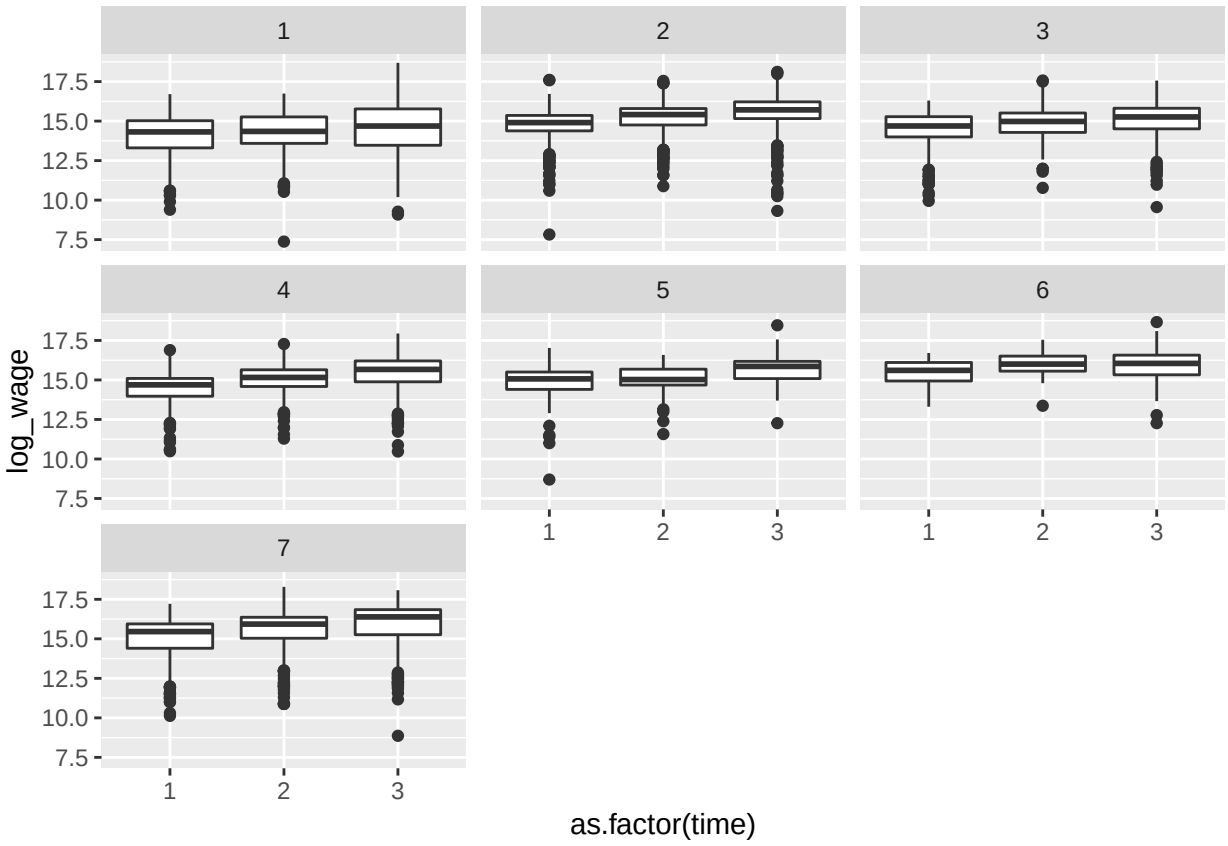


Log wages by Class and Time

```
wage_job_dat <- wage_job_dat %>%
  mutate(time_job = as.factor(paste0(time,"-",job)))
ggplot(wage_job_dat, aes(x = job, y = log_wage)) + geom_boxplot() +
  facet_wrap(~time)
```

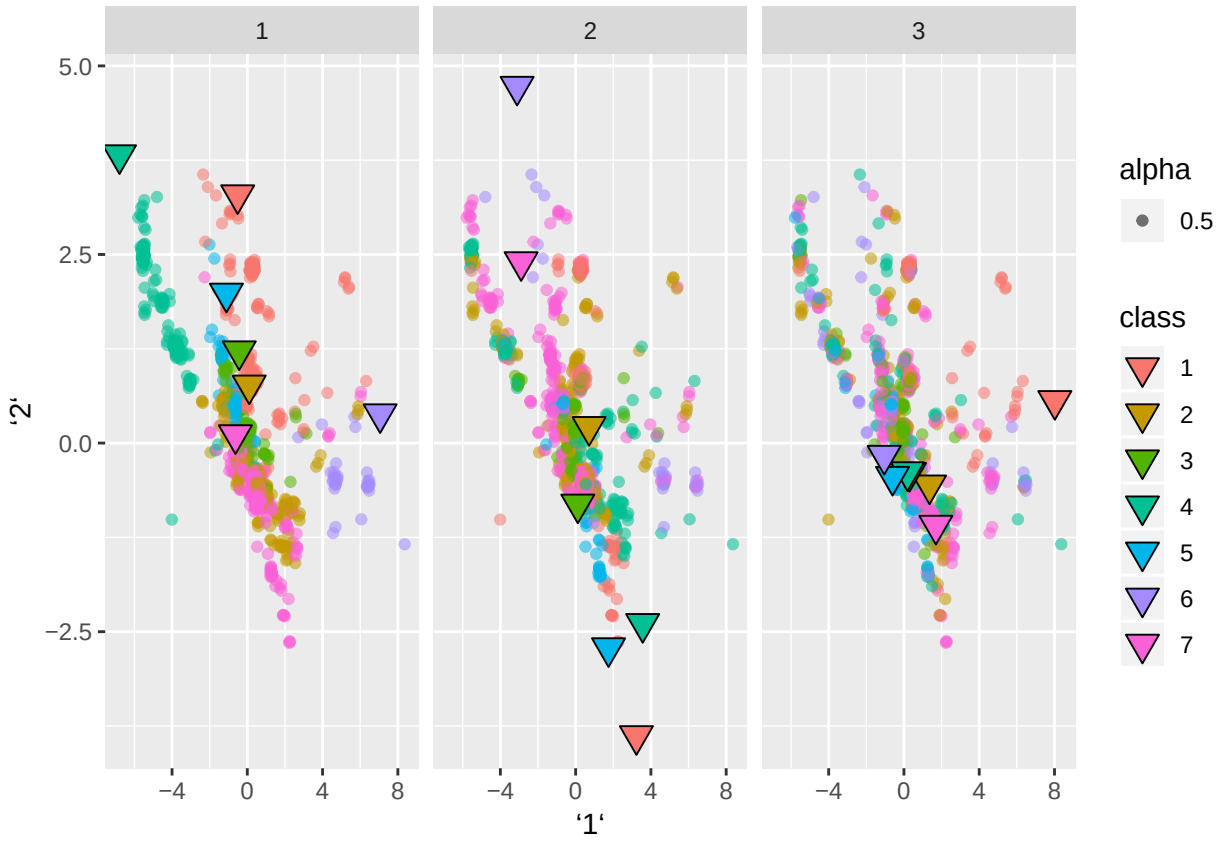



```
ggplot(wage_job_dat, aes(x = as.factor(time), y = log_wage)) + geom_boxplot() +
  facet_wrap(~job)
```



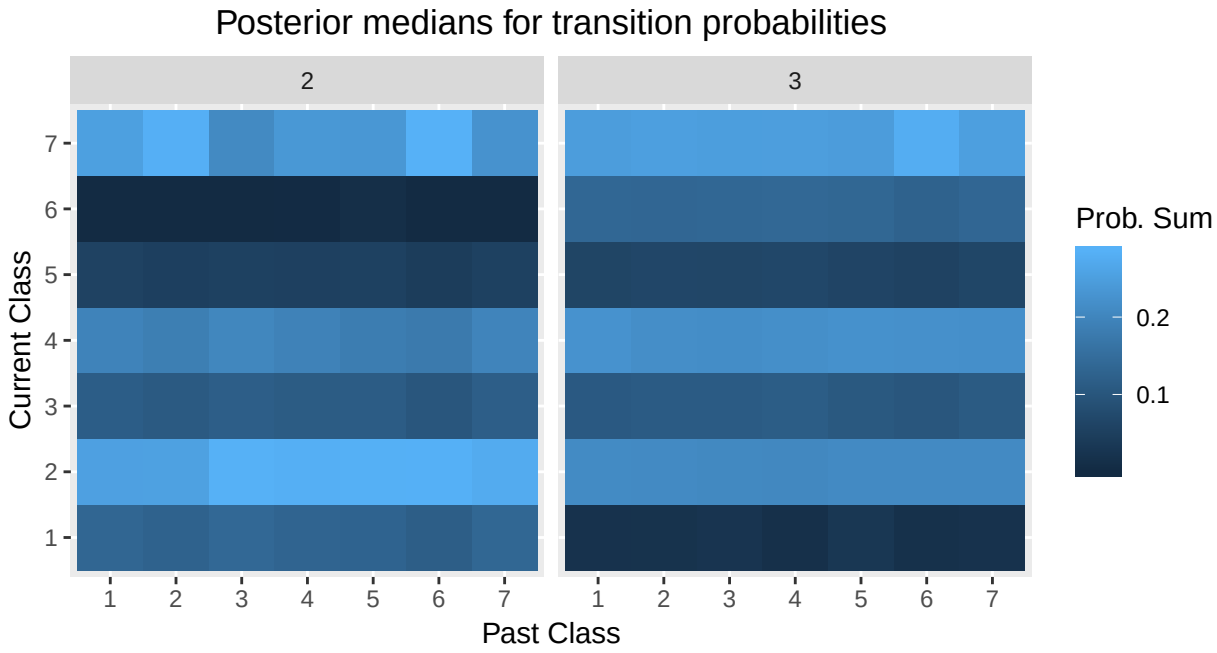
Plot latent space positions

```
med_z <- data.frame(par = rownames(fit_summ), med = fit_summ[,6]) %>%
  filter(grepl("z\\[", par)) %>%
  cSplit("par",direction = "wide") %>%
  mutate(worker = stri_extract_first_regex(par_1,"[0-9]+"),
         dim = stri_extract_first_regex(par_2,"[0-9]+")) %>%
  select(-c("par_1","par_2")) %>%
  tidyr::pivot_wider(names_from = dim, values_from = med) %>%
  mutate(job1 = job[,1],
         job2 = job[,2],
         job3 = job[,3]) %>%
  tidyr::pivot_longer(cols = c(job1,job2,job3),names_to = "time",names_prefix = "job") %>%
  mutate(class = as.factor(value))
med_w <- data.frame(par = rownames(fit_summ), med = fit_summ[,6]) %>%
  filter(grepl("w\\[", par)) %>%
  cSplit("par",direction = "wide") %>%
  mutate(class = stri_extract_first_regex(par_1,"[0-9]+"),
         time = stri_extract_first_regex(par_2,"[0-9]+"),
         dim = stri_extract_first_regex(par_3,"[0-9]+")) %>%
  select(-c("par_1","par_2","par_3"))%>%
  tidyr::pivot_wider(names_from = dim, values_from = med)
ggplot() +  geom_point(data = med_z, aes(x = `1`, y = `2`, color = class, alpha = 0.5)) +
  geom_point(data = med_w, aes(x = `1`, y = `2`, fill = class),
             shape = 25, size = 4, color = "black")  +
  facet_wrap(~time)
```



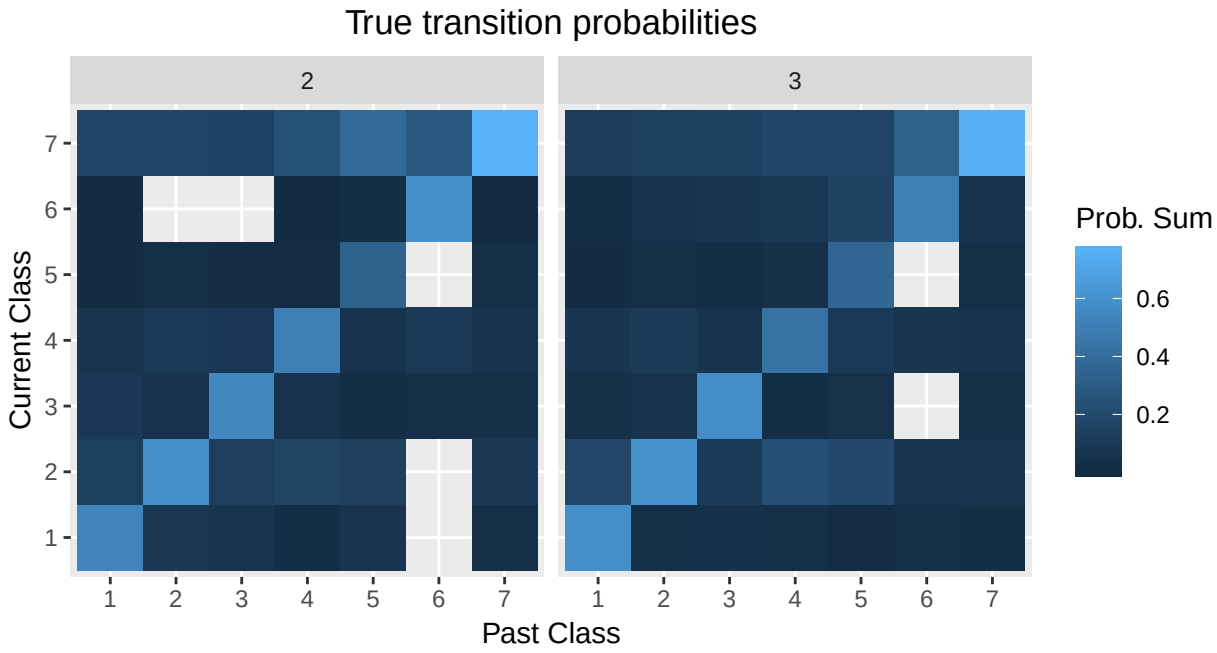
Probability transition plots - Posterior medians

```
med_p <- data.frame(par = rownames(fit_summ), med = fit_summ[,6]) %>%
  filter(grepl("p\\[", par)) %>%
  cSplit("par",direction = "wide") %>%
  mutate(worker = stri_extract_first_regex(par_1,"[0-9]+"),
         time = stri_extract_first_regex(par_2,"[0-9]+"),
         class = stri_extract_first_regex(par_3,"[0-9]+")) %>%
  select(-c("par_1","par_2","par_3")) %>%
  filter(time %in% c(2,3)) %>%
  mutate(past_class = c(rep(job[,2],each=7),rep(job[,3],each=7)))
trans_probs <- med_p %>% group_by(time,past_class) %>%
  mutate(tot_sum = sum(med),
         prop = med/tot_sum) %>%
  group_by(time,class,past_class) %>%
  summarise(prob_sum = sum(prop)) %>%
  arrange(past_class)
ggplot(trans_probs, aes(x = as.factor(past_class), y = as.factor(class), fill = prob_sum)) +
  geom_tile() + facet_grid(cols = vars(time)) +
  xlab("Past Class") + ylab("Current Class") + labs(fill='Prob. Sum') +
  ggtitle("Posterior medians for transition probabilities") +
  theme(aspect.ratio = 1) + theme(plot.title = element_text(hjust = 0.5))
```



Probability transition plots - Data

```
real_trans <- data.frame(past = c(job[,1],job[,2]), current = c(job[,2],job[,3]),
                         time = c(rep(2,N),rep(3,N))) %>%
  group_by(past,time) %>%
  mutate(tot_sum = n(),
         prop = 1/tot_sum) %>%
  group_by(past,current,time) %>%
  summarise(prob_sum = sum(prop)) %>%
  arrange(past,time)
ggplot(real_trans, aes(x = as.factor(past), y = as.factor(current), fill = prob_sum)) +
  geom_tile() + facet_grid(cols = vars(time)) +
  xlab("Past Class") + ylab("Current Class") + labs(fill='Prob. Sum') +
  ggtitle("True transition probabilities") +
  theme(aspect.ratio = 1) + theme(plot.title = element_text(hjust = 0.5))
```


True transition probabilities
2
3
Current Class
Past Class
1
2
3
4
5
6
7
Prob. Sum
0.6
0.4
0.2