

eda

Fanyu, Ryan, Serena

2024-02-19

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.1      v readr      2.1.4
## v forcats    1.0.0      v stringr   1.5.0
## v ggplot2    3.4.3      v tibble    3.2.1
## v lubridate  1.9.2      v tidyr     1.3.0
## v purrr      1.0.1
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lme4)
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 4.2.3
```

```
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
library(nlme)
```

```
##
## Attaching package: 'nlme'
##
## The following object is masked from 'package:lme4':
##
##     lmList
##
## The following object is masked from 'package:dplyr':
##
##     collapse
```

```
library(lattice)
library(ggplot2)
library(forcats)
```

```
source(knitr::purl("P9185_project1.Rmd", quiet=TRUE))
```

```
## This is DHARMA 0.4.6. For overview type '?DHARMA'. For recent changes, type news(package = 'DHARMA')
```

```
##
```

```
## Attaching package: 'kableExtra'
```

```
## The following object is masked from 'package:dplyr':
```

```
##
```

```
## group_rows
```

```
## Joining with 'by = join_by(ptid, period, drug)'
```

```
## Joining with 'by = join_by(ptid, drug)'
```

```
## Joining with 'by = join_by(ptid, drug)'
```

```
## 'summarise()' has grouped output by 'ptid'. You can override using the
```

```
## '.groups' argument.
```

```
## Joining with 'by = join_by(ptid, period)'
```

```
## Joining with 'by = join_by(ptid, period)'
```

```
## Joining with 'by = join_by(ptid)'
```

```
## Joining with 'by = join_by(ptid)'
```

```
## Registered S3 method overwritten by 'webshot2': method from print.webshot
```

```
## webshot
```

```
## Joining with 'by = join_by(ptid)'
```

```
## Joining with 'by = join_by(ptid)'
```

```
## boundary (singular) fit: see help('isSingular')
```

```
## Joining with 'by = join_by(ptid)'
```

```
## Joining with 'by = join_by(ptid)'
```

```
## Attaching package: 'table1'
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## units, units<-
```

```
vload_df <- baseline.dat %>%
  pivot_longer(cols = contains("viral"), names_to = "type", values_to = "value") %>%
  mutate(obs_num = as.numeric(str_sub(type,-1,-1)),
         measure = str_sub(type,1,1),
         type = case_when(obs_num == 0 ~ "baseline",
                          obs_num %% 2 == 0 ~ "washout",
                          .default = "after"),
         period = case_when(
           obs_num <= 1 ~ 1,
           obs_num %in% c(2,3) ~ 2,
           obs_num %in% c(4,5,6) ~ 3,
           #washout in the end
           .default = 4),
         sequence = paste0(
           str_sub(period,1,-1,-1),
```

```

      str_sub(period2,-1,-1),
      str_sub(period3,-1,-1))) %>%
left_join(endpoints.AE %>%select(-week) %>% distinct(), by = c("ptid", "period", "sequence")) %>%
mutate(measure = if_else(measure == "b", "blood", "skin")) %>%
drop_na()

```

```

## Warning in left_join(., endpoints.AE %>% select(-week) %>% distinct(), by = c("ptid", : Detected an
## i Row 37 of 'x' matches multiple rows in 'y'.
## i Row 1 of 'y' matches multiple rows in 'x'.
## i If a many-to-many relationship is expected, set 'relationship =
##   "many-to-many"' to silence this warning.

```

```

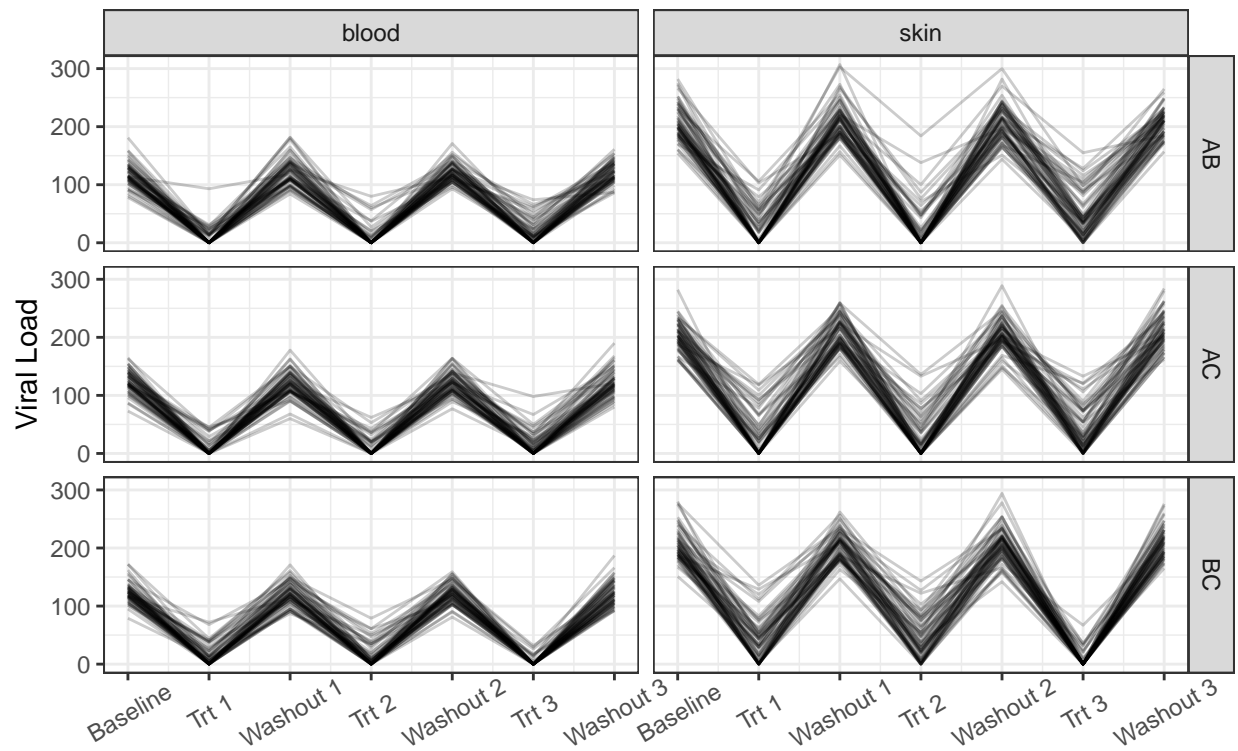
vload_df$seq2 = factor(vload_df$seq2)

#levels(vload_df$seq2) <- c("0" = 'rho[AB]', "1" = 'rho[AC]', "2" = 'rho[BC]')

# maybe not including washout 3
vload_df %>%
  ggplot(aes(x = obs_num, y = value, group = ptid)) +
  geom_line(alpha = 0.2, linewidth = 0.5)+
  #facet_grid(seq2~measure,labeller = label_parsed)+
  facet_grid(seq2~measure)+
  theme_bw()+
  ylab("Viral Load")+
  xlab("")+
  scale_x_continuous(breaks=seq(0,6),
                     labels=c("0" = "Baseline", "1" = "Trt 1",
                               "2" = "Washout 1", "3" = "Trt 2",
                               "4" = "Washout 2", "5" = "Trt 3",
                               "6" = "Washout 3"))+
  theme(axis.text.x = element_text(angle = 30, vjust = 0.7))+
  ggtitle("Observed Viral Load & Carryover Effects")

```

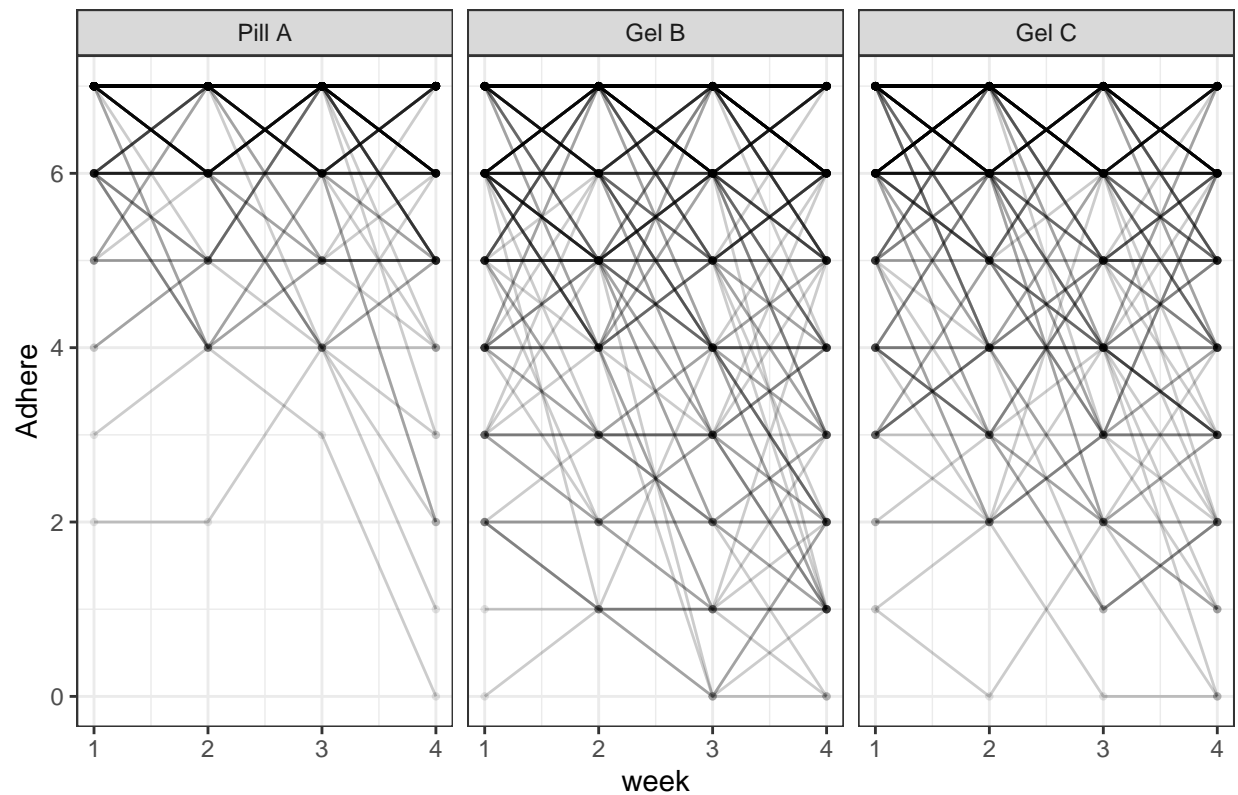
Observed Viral Load & Carryover Effects



*#not really plotting observed rhos, just facet by seq2
#maybe need to change facet titles*

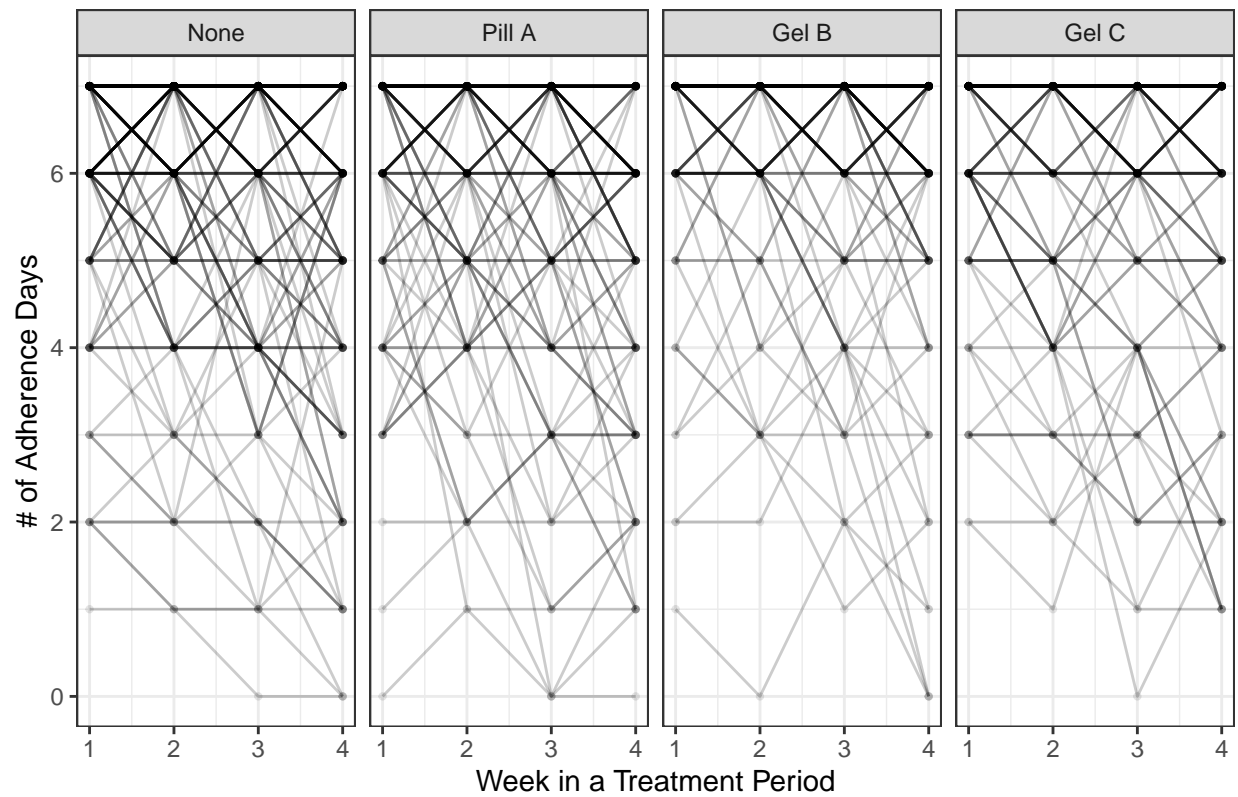
```
endpoints.Adhere %>%
  # each line is adherence of a drug taken by a subject
  ggplot(aes(x = week, y = Adhere, group = interaction(ptid, drug))) +
  geom_line(alpha = 0.2) +
  geom_point(alpha = 0.1, size = 0.8) +
  facet_grid(.~drug) +
  theme_bw() +
  ggtitle("Adherence by Drug Over Weeks")
```

Adherence by Drug Over Weeks



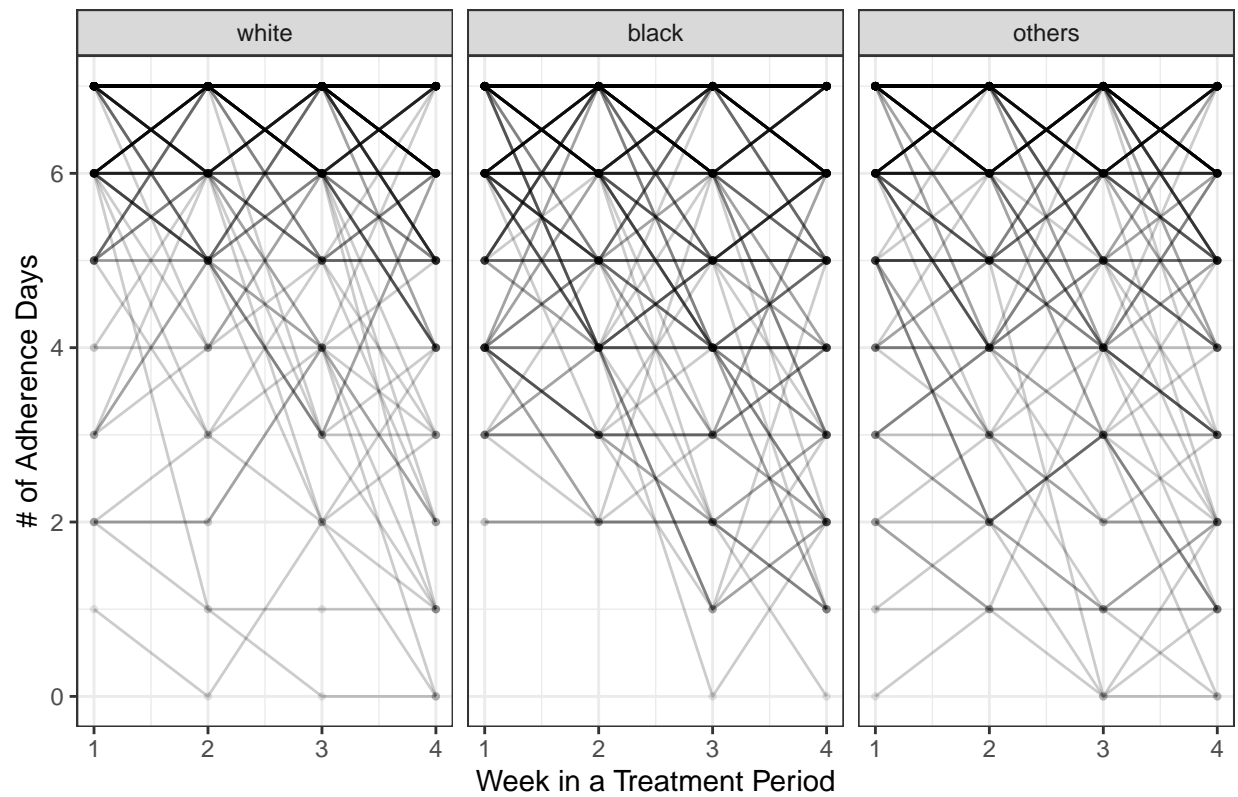
```
endpoints.Adhere %>%
  ggplot(aes(x = week, y = Adhere, group = interaction(ptid, drug))) +
  geom_line(alpha = 0.2) +
  geom_point(alpha = 0.1, size = 0.8) +
  facet_grid(.~factor(drug_lag, levels=c('None', 'Pill A', 'Gel B', 'Gel C')))+
  theme_bw() +
  ylab("# of Adherence Days")+
  xlab("Week in a Treatment Period")+
  ggtitle("Weekly Adherence by Carryover Effects")
```

Weekly Adherence by Carryover Effects

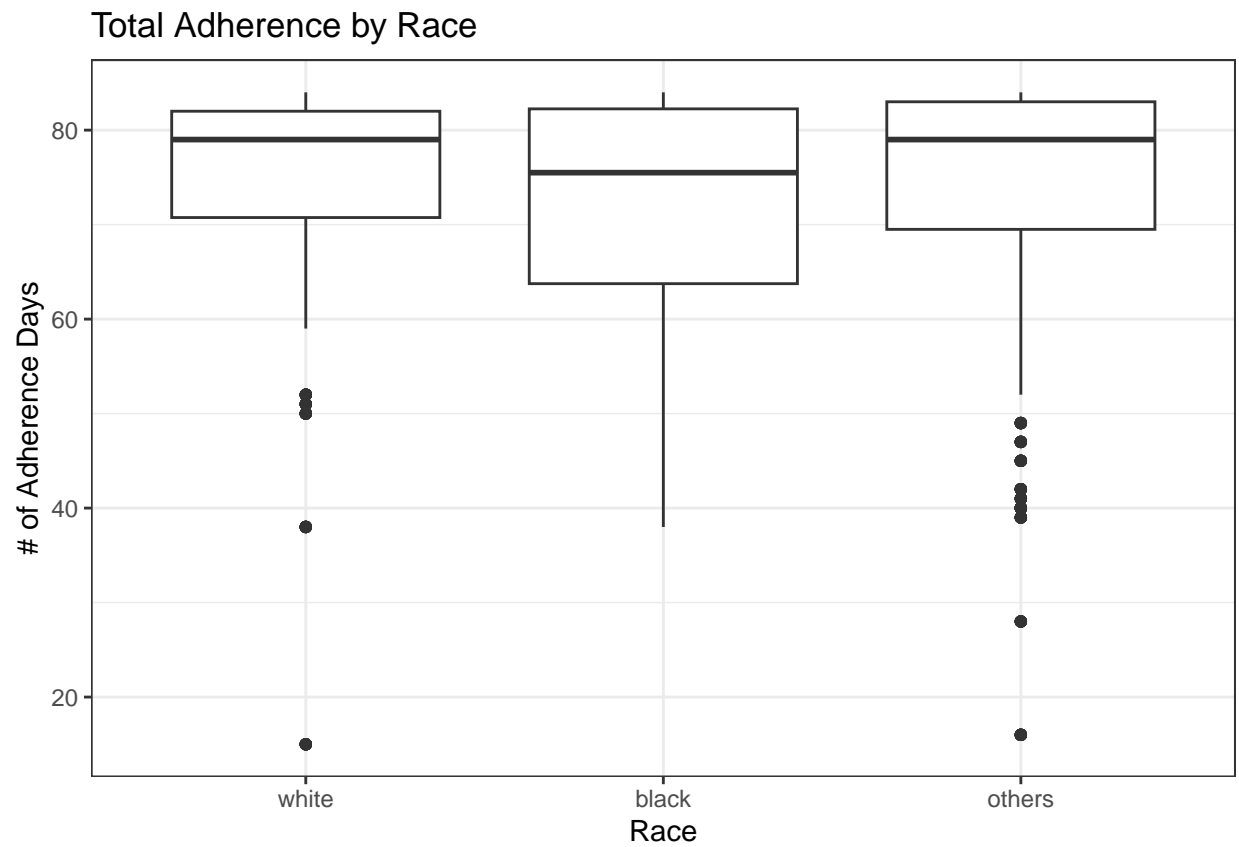


```
endpoints.Adhere %>%
  ggplot(aes(x = week, y = Adhere, group = interaction(ptid, drug))) +
  geom_line(alpha = 0.2) +
  geom_point(alpha = 0.1, size = 0.8) +
  facet_grid(.~race)+
  theme_bw() +
  ylab("# of Adherence Days")+
  xlab("Week in a Treatment Period")+
  ggtitle("Weekly Adherence by Race")
```

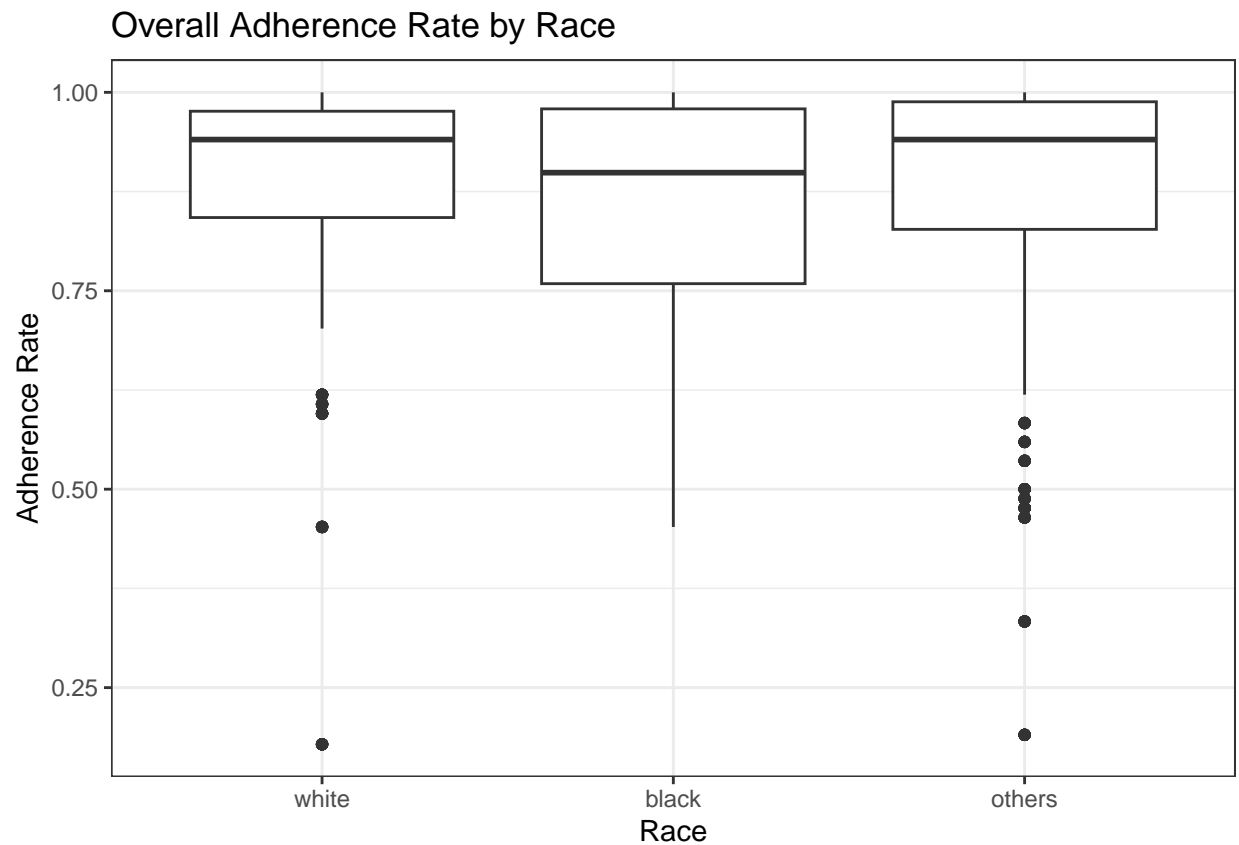
Weekly Adherence by Race



```
endpoints.Adhere %>%
  ggplot(aes(y = Adhere_total, x = race))+
  geom_boxplot()+
  theme_bw()+
  ylab("# of Adherence Days")+
  xlab("Race")+
  ggtitle("Total Adherence by Race")
```

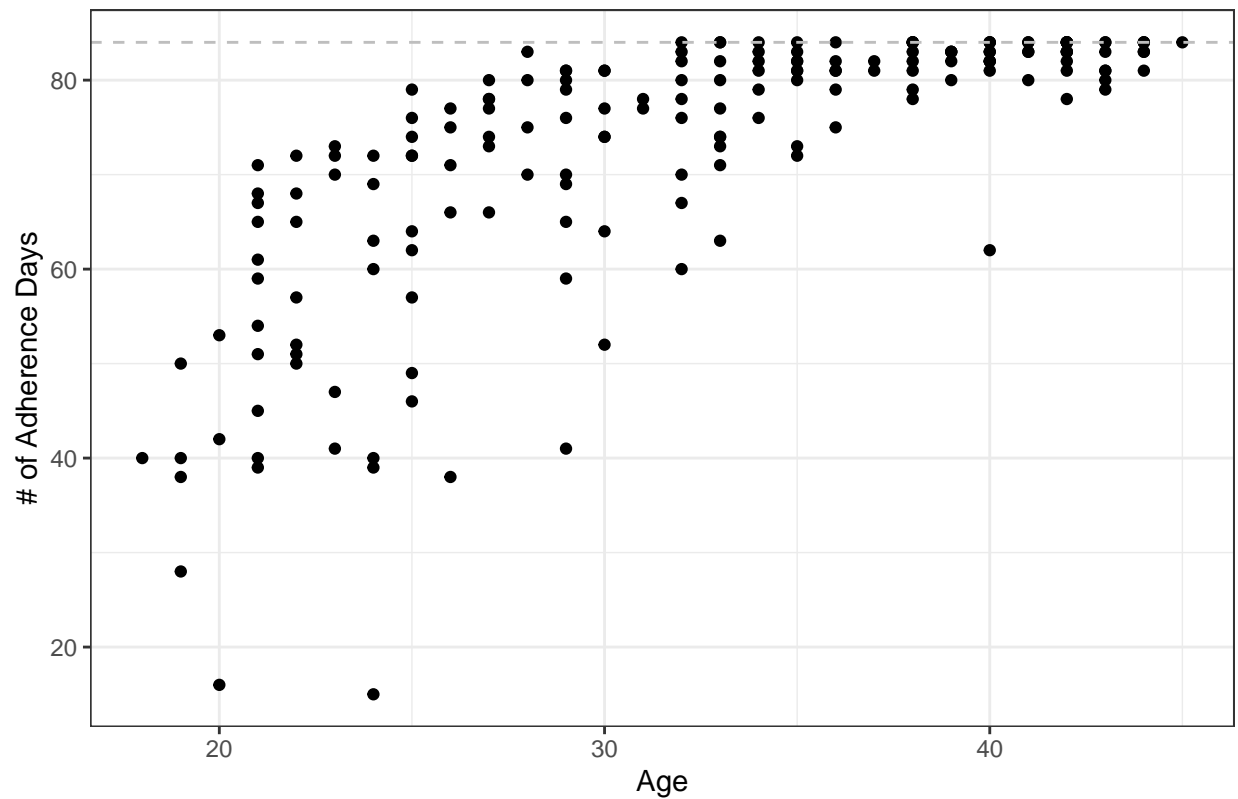


```
endpoints.Adhere %>%  
  ggplot(aes(y = Adhere_total/84, x = race))+  
  geom_boxplot()+  
  theme_bw()+  
  ylab("Adherence Rate")+  
  xlab("Race")+  
  ggtitle("Overall Adherence Rate by Race")
```

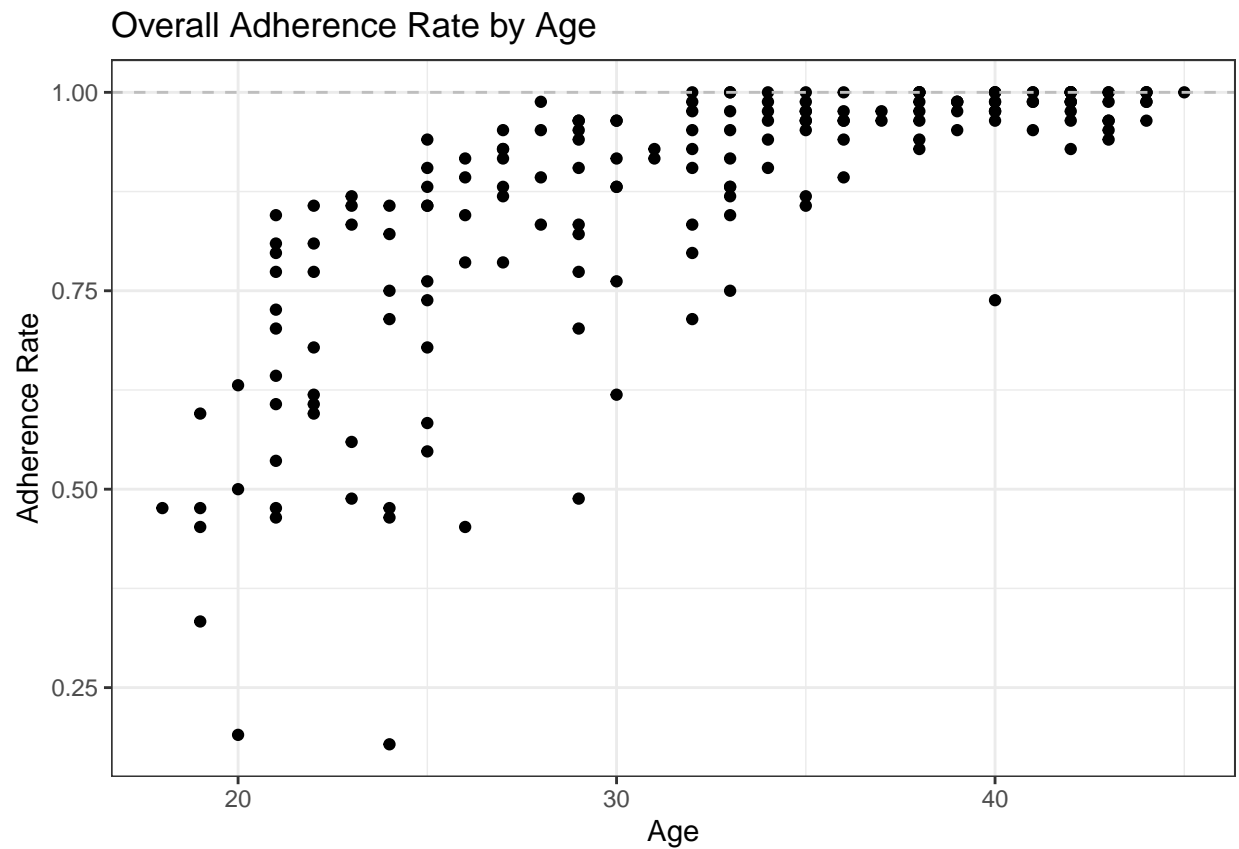



```
Adhere.demo %>%
  ggplot(aes(y = Adhere_total, x = age))+
  geom_point()+
  geom_hline(yintercept = 84, linetype = 2, color = "grey")+
  theme_bw()+
  ylab("# of Adherence Days")+
  xlab("Age")+
  ggtitle("Total Adherence by Age")
```

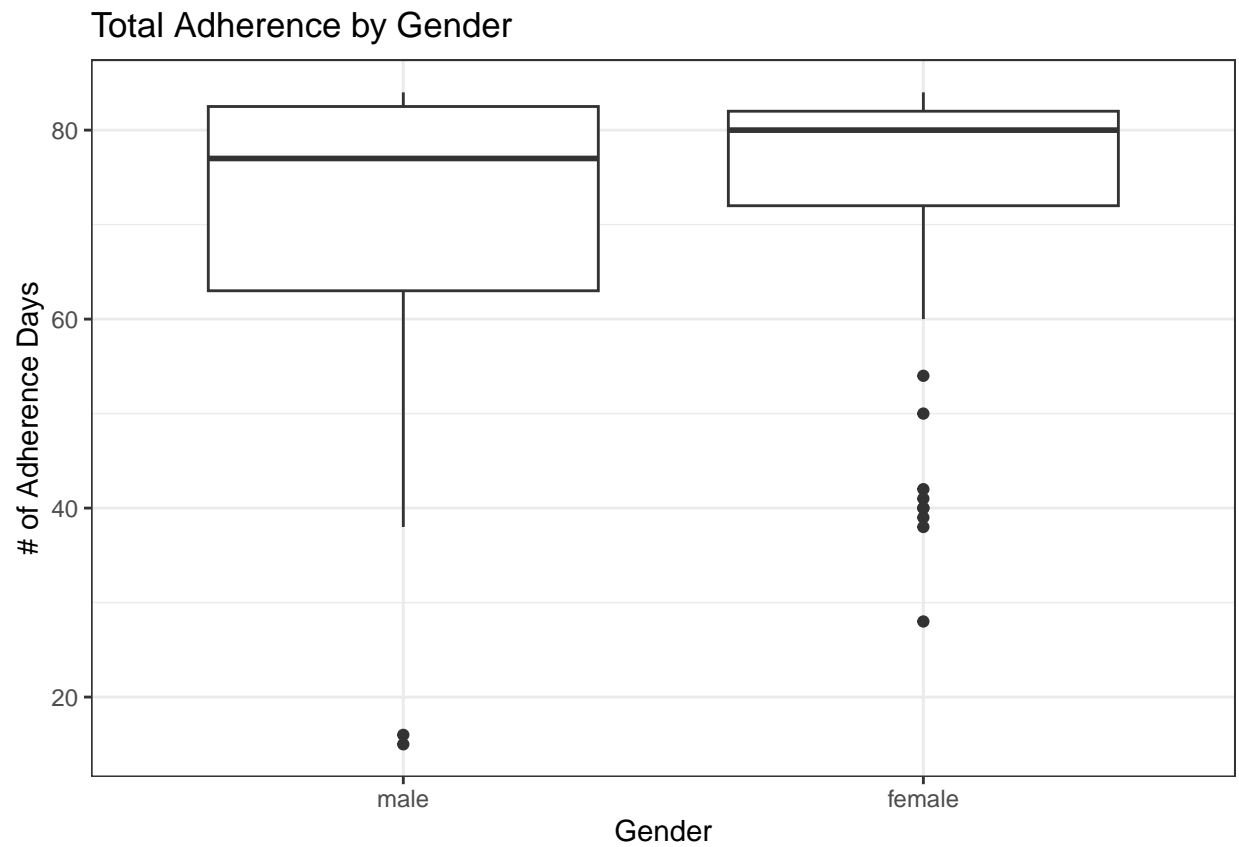
Total Adherence by Age



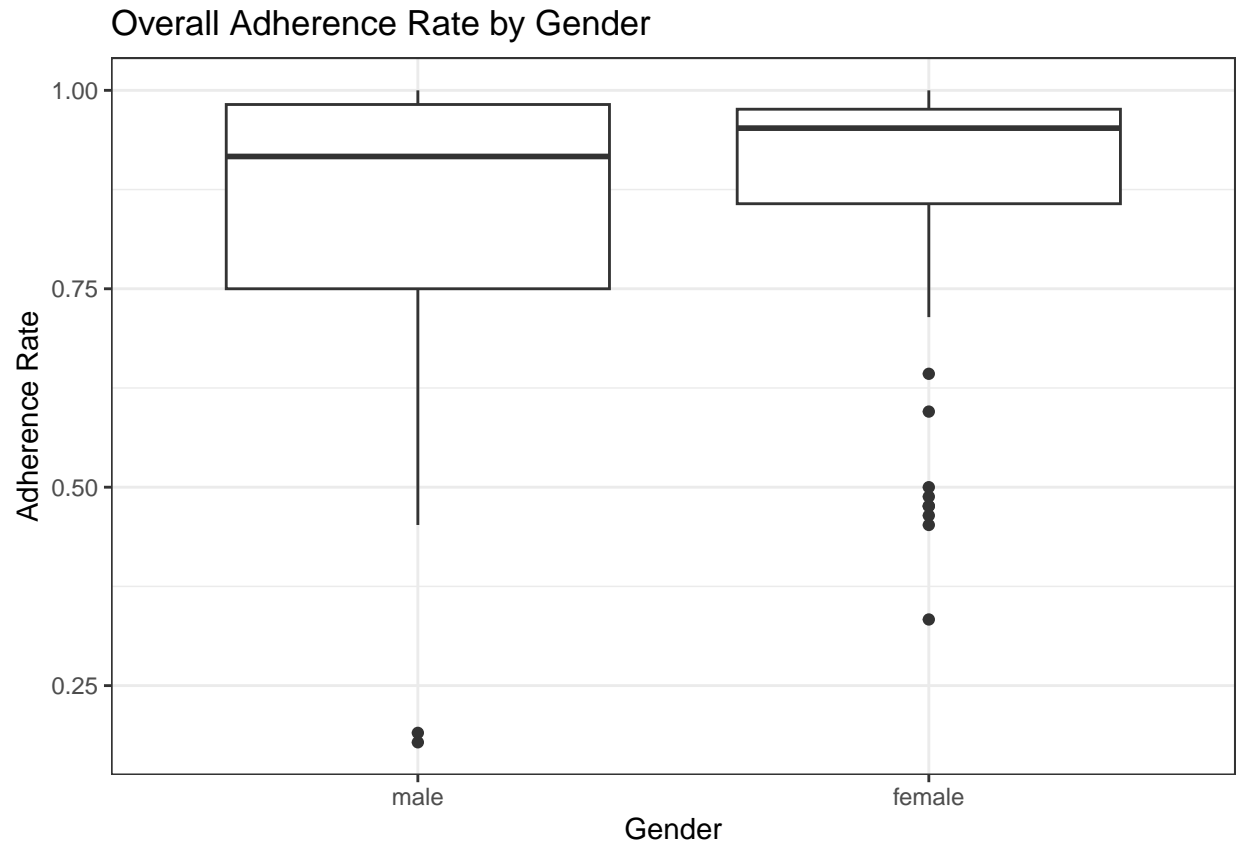
```
Adhere.demo %>%
  ggplot(aes(y = Adhere_total/84, x = age))+
  geom_point()+
  geom_hline(yintercept = 1, linetype = 2, color = "grey")+
  theme_bw()+
  ylab("Adherence Rate")+
  xlab("Age")+
  ggtitle(label = "Overall Adherence Rate by Age")
```



```
Adhere.demo %>%
  ggplot(aes(y = Adhere_total, x = gender))+
  geom_boxplot()+
  theme_bw()+
  ylab("# of Adherence Days")+
  xlab("Gender")+
  ggtitle("Total Adherence by Gender")
```



```
Adhere.demo %>%  
  ggplot(aes(y = Adhere_total/84, x = gender))+  
  geom_boxplot()+  
  theme_bw()+  
  ylab("Adherence Rate")+  
  xlab("Gender")+  
  ggtitle("Overall Adherence Rate by Gender")
```

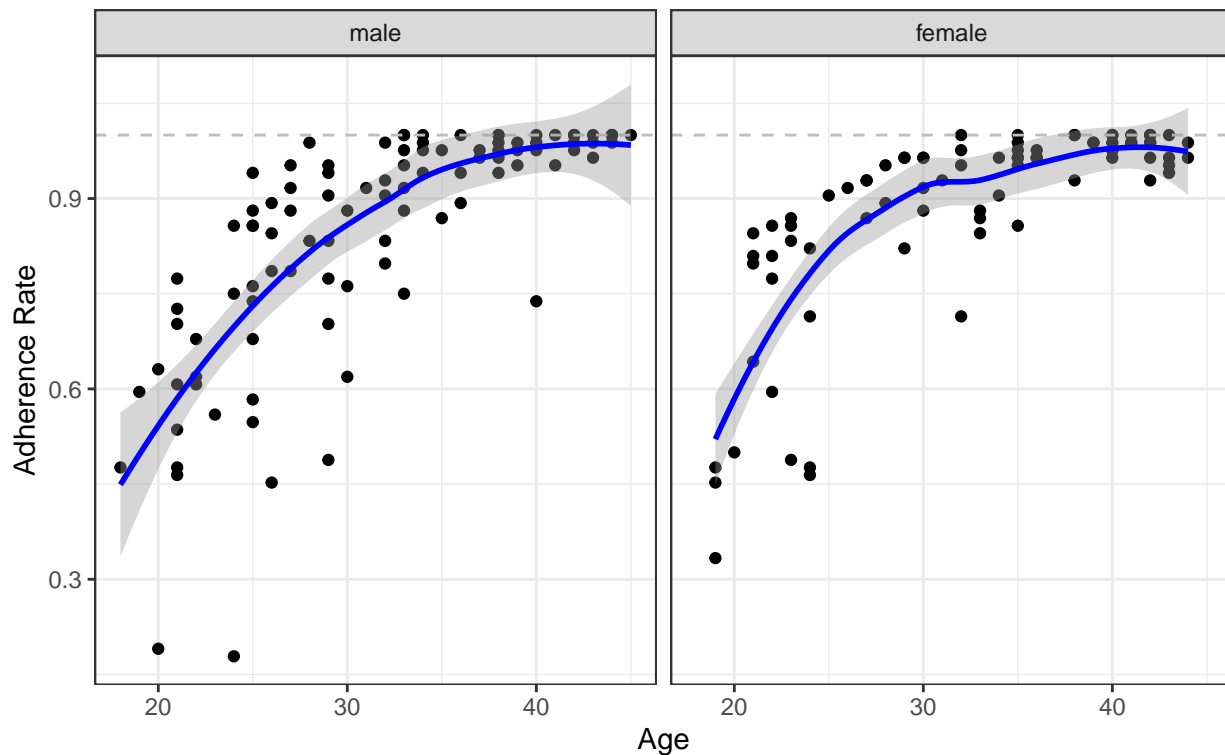


```
Adhere.demo %>%
  ggplot(aes(y = Adhere_total/84, x = age))+
  geom_point()+
  geom_hline(yintercept = 1, linetype = 2, color = "grey")+
  geom_smooth(color="blue", method="loess")+
  facet_grid(.~gender)+
  theme_bw()+
  ylab("Adherence Rate")+
  xlab("Age")+
  ggtitle(label = "Overall Adherence Rate by Age & Gender", subtitle = "With Loess Smoothing and 95% CI")
```

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

Overall Adherence Rate by Age & Gender

With Loess Smoothing and 95% CI

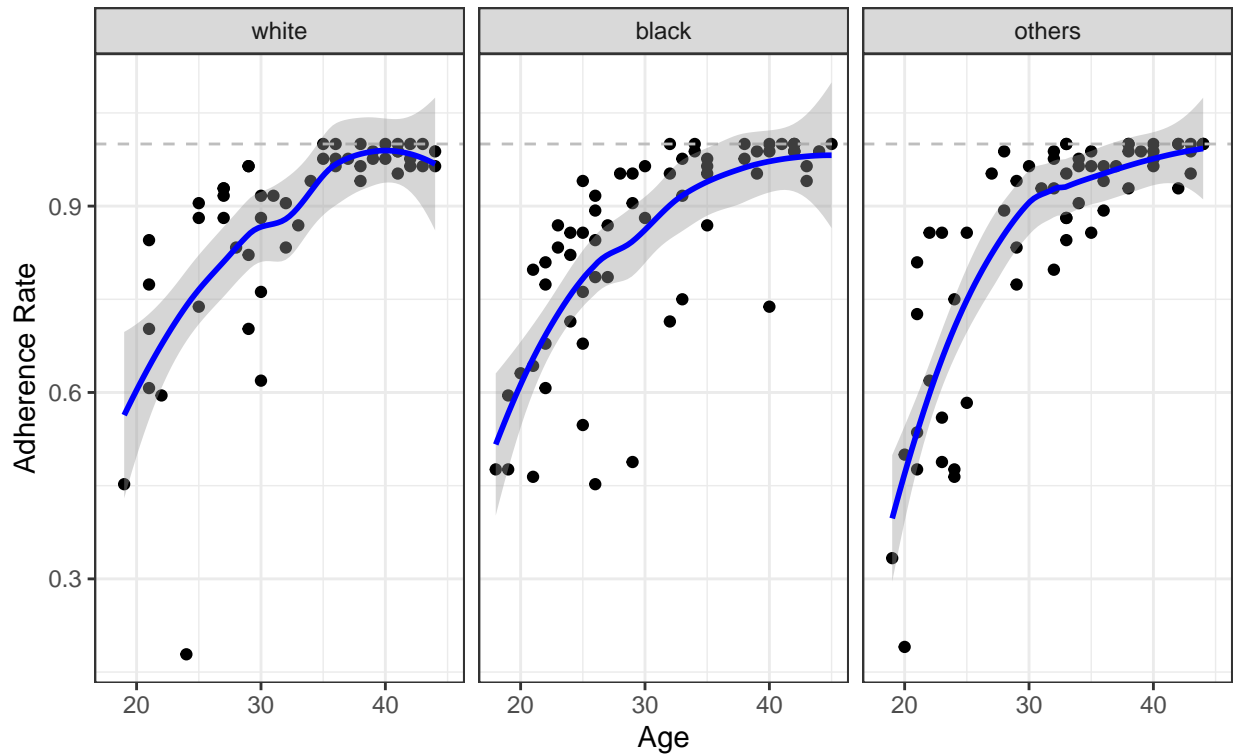


```
Adhere.demo %>%
  ggplot(aes(y = Adhere_total/84, x = age))+
  geom_point()+
  geom_hline(yintercept = 1, linetype = 2, color = "grey")+
  geom_smooth(color="blue", method="loess")+
  facet_grid(.~race)+
  theme_bw()+
  ylab("Adherence Rate")+
  xlab("Age")+
  ggtitle(label = "Overall Adherence Rate by Age & Race", subtitle = "With Loess Smoothing and 95% CI")
```

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

Overall Adherence Rate by Age & Race

With Loess Smoothing and 95% CI

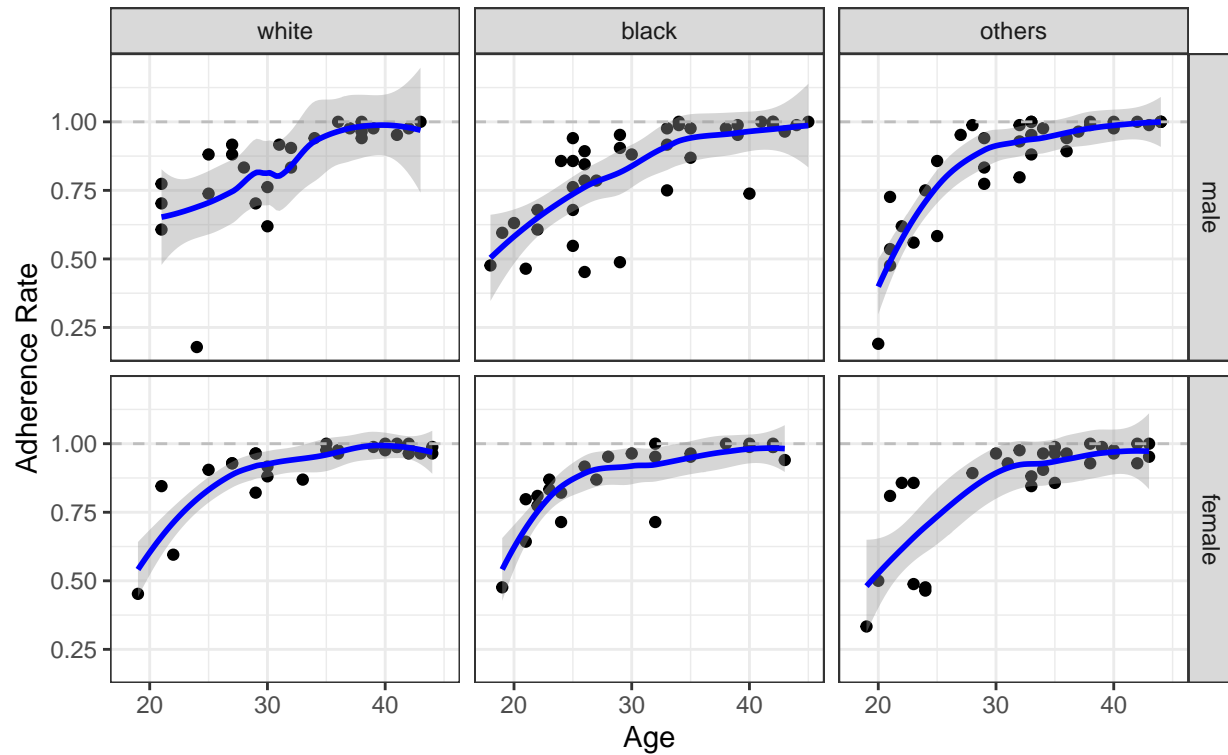


```
Adhere.demo %>%
  ggplot(aes(y = Adhere_total/84, x = age))+
  geom_point()+
  geom_hline(yintercept = 1, linetype = 2, color = "grey")+
  geom_smooth(color="blue", method="loess")+
  facet_grid(gender~race)+
  theme_bw()+
  ylab("Adherence Rate")+
  xlab("Age")+
  ggtitle(label = "Overall Adherence Rate by Age & Race & Gender", subtitle = "With Loess Smoothing and 95% CI")
```

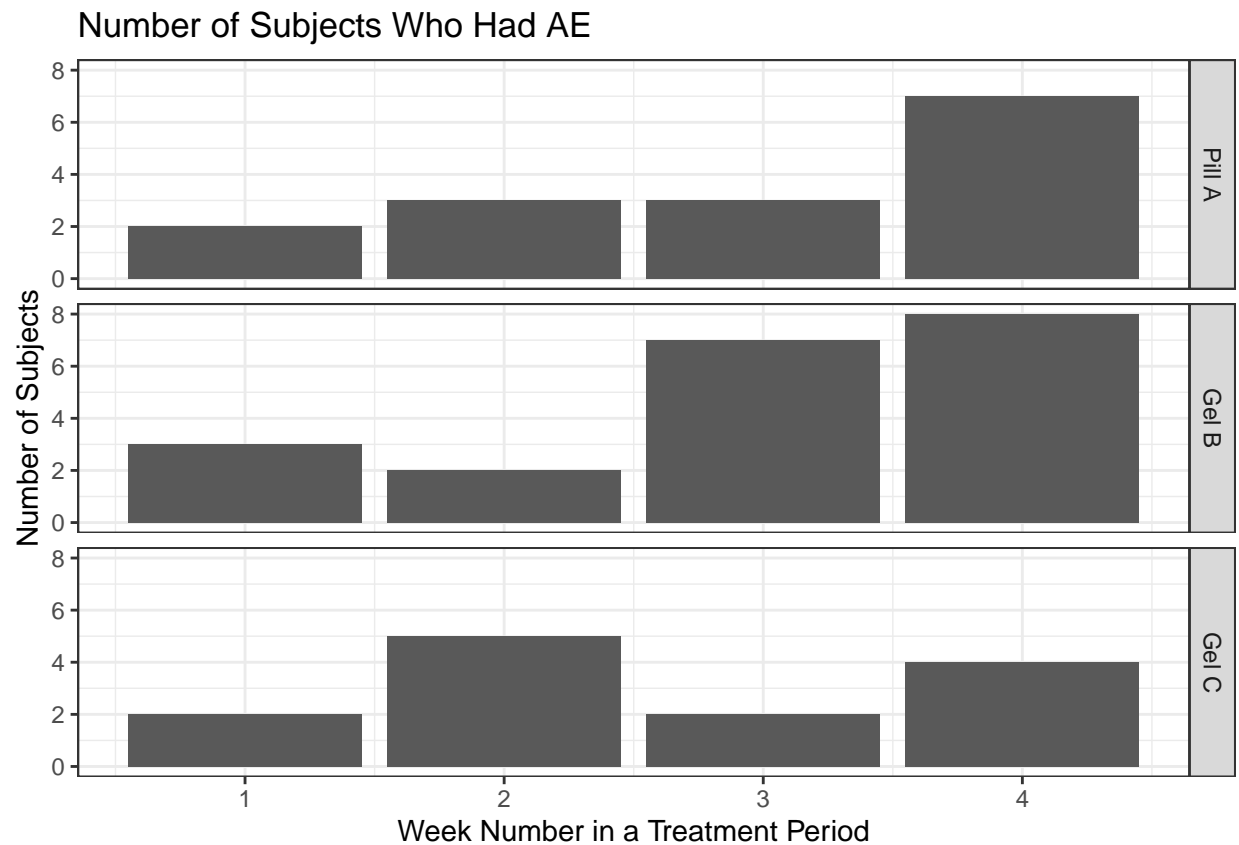
```
## 'geom_smooth()' using formula = 'y ~ x'
```

Overall Adherence Rate by Age & Race & Gender

With Loess Smoothing and 95% CI

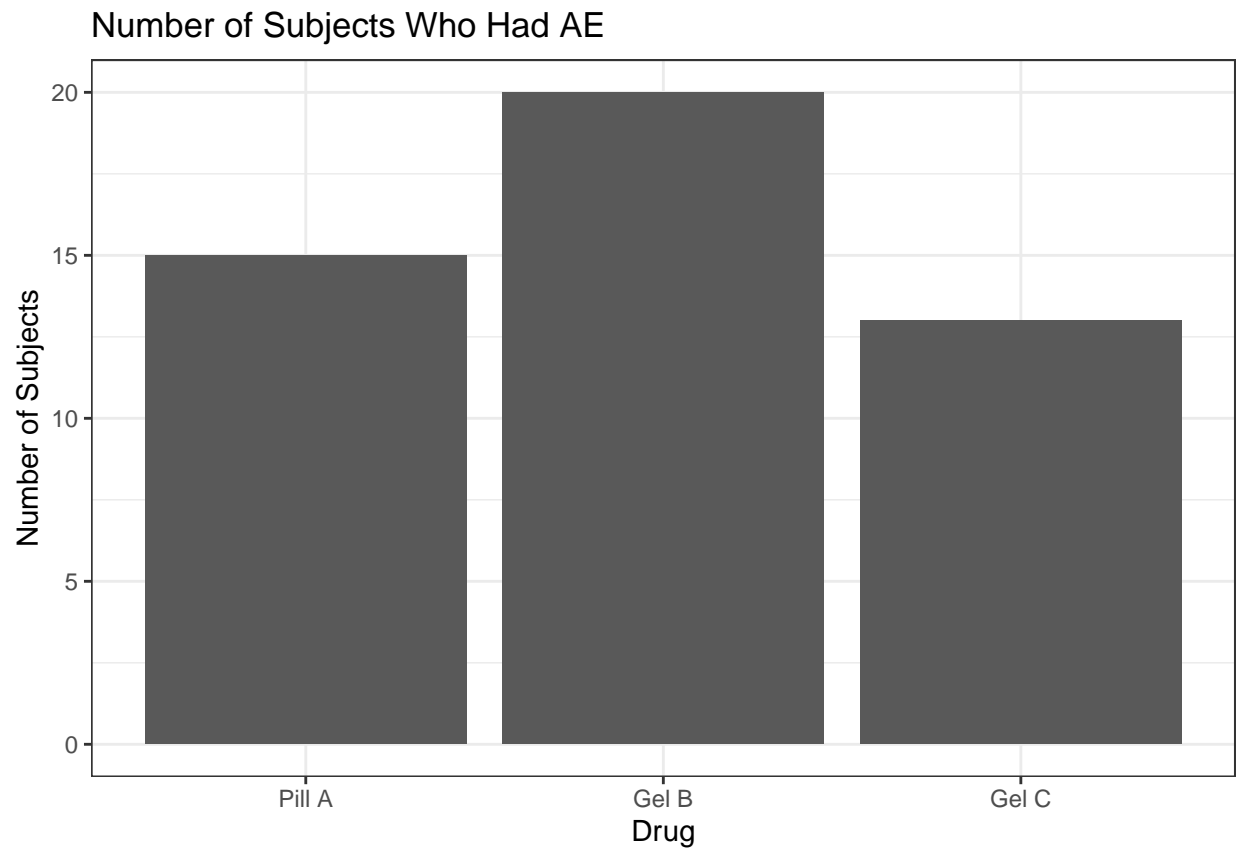


```
endpoints.AE %>%
  filter(AE ==1) %>%
  group_by(drug, week) %>%
  ggplot(aes(x=week))+
  geom_bar()+
  facet_grid(factor(drug, levels=c('Pill A', 'Gel B', 'Gel C'))~.)+
  theme_bw()+
  ylab("Number of Subjects")+
  xlab("Week Number in a Treatment Period")+
  ggtitle("Number of Subjects Who Had AE")
```

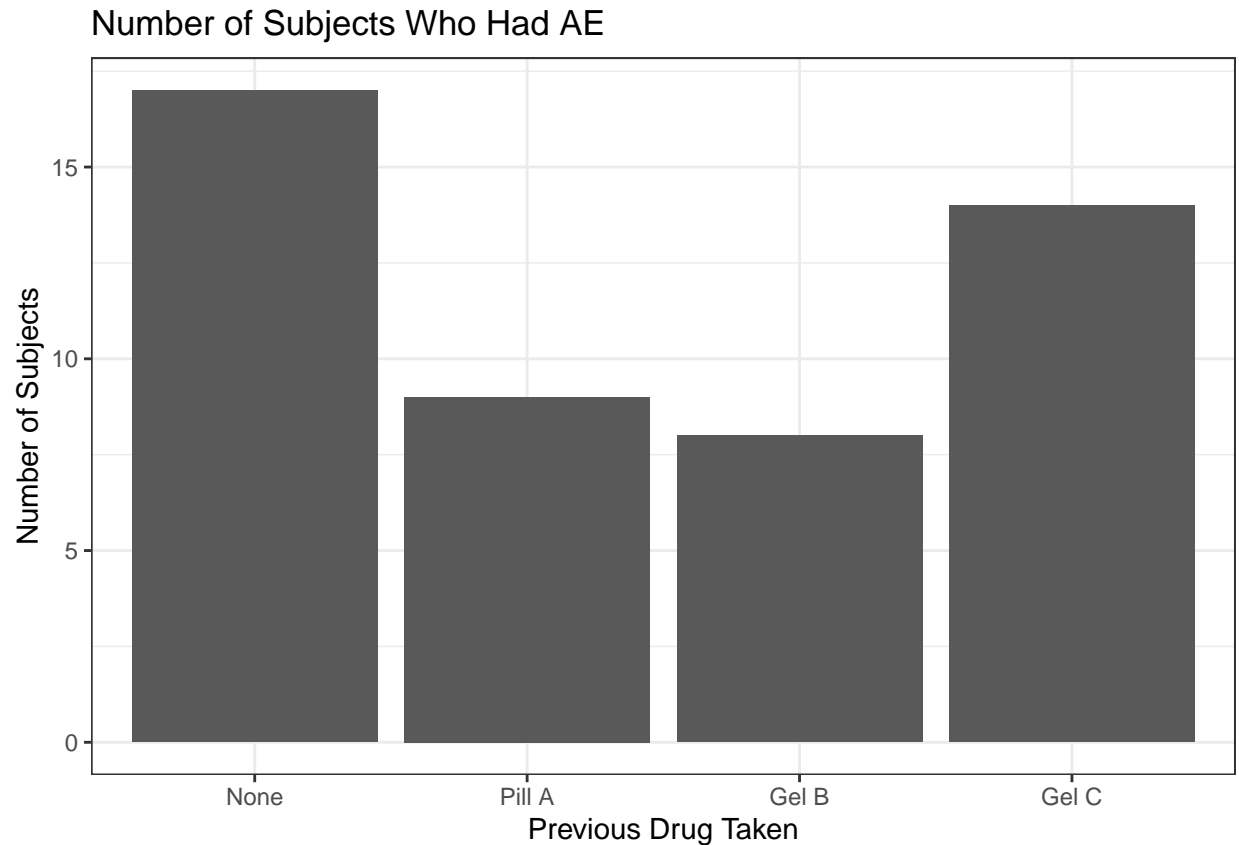



it's a balanced design, so can compare count value

```
endpoints.AE %>%
  filter(AE ==1) %>%
  ggplot(aes(x=factor(drug, levels=c('Pill A', 'Gel B', 'Gel C')))) +
  geom_bar() +
  theme_bw() +
  ylab("Number of Subjects") +
  xlab("Drug") +
  ggtitle("Number of Subjects Who Had AE")
```

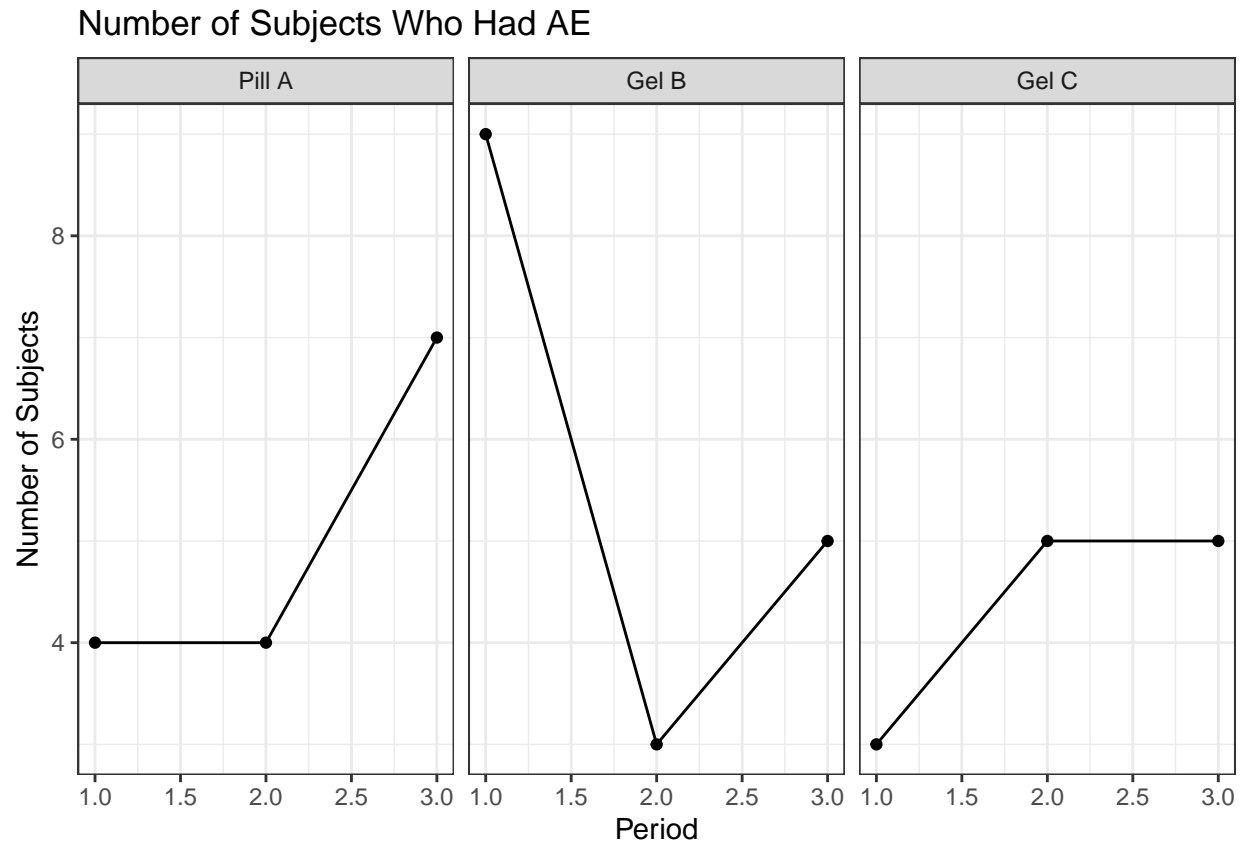


```
endpoints.AE %>%  
  filter(AE ==1) %>%  
  ggplot(aes(x=factor(drug_lag, levels=c('None', 'Pill A', 'Gel B', 'Gel C'))))+  
  geom_bar()+  
  theme_bw()+  
  ylab("Number of Subjects")+  
  xlab("Previous Drug Taken")+  
  ggtitle("Number of Subjects Who Had AE")
```



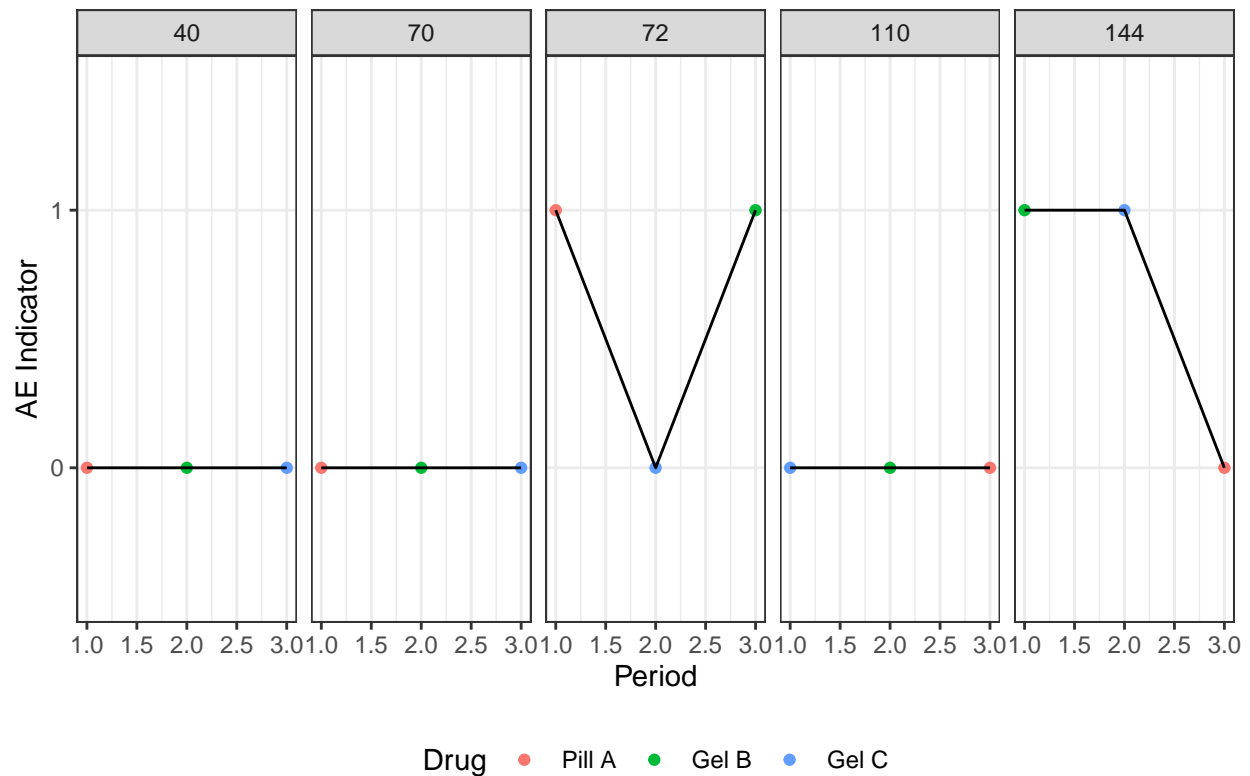
```
endpoints.AE %>%  
  filter(AE == 1) %>%  
  group_by(period, drug) %>%  
  summarise(count = n_distinct(ptid)) %>%  
  ggplot(aes(x=period, y=count))+  
  geom_point()+  
  geom_line()+  
  facet_grid(.~factor(drug, levels=c('Pill A', 'Gel B', 'Gel C')))+  
  theme_bw()+  
  ylab("Number of Subjects")+  
  xlab("Period")+  
  ggtitle("Number of Subjects Who Had AE")
```

'summarise()' has grouped output by 'period'. You can override using the
'.groups' argument.



```
set.seed(202)
endpoints.AE.weeksum %>%
  filter(ptid %in% sample.int(180, 5))%>%
  ggplot(aes(x=period, y=AE_ind, group =interaction(ptid, drug)))+
  geom_point(aes(color=factor(drug, levels=c('Pill A','Gel B','Gel C'))))+
  geom_line(aes(group = ptid))+
  facet_grid(.~ptid)+
  theme_bw()+
  ylab("AE Indicator")+
  xlab("Period")+
  ggtitle("AE Record of 5 Randomly Selected Subjects")+
  theme(legend.position = "bottom")+
  guides(color=guide_legend(title="Drug"))
```

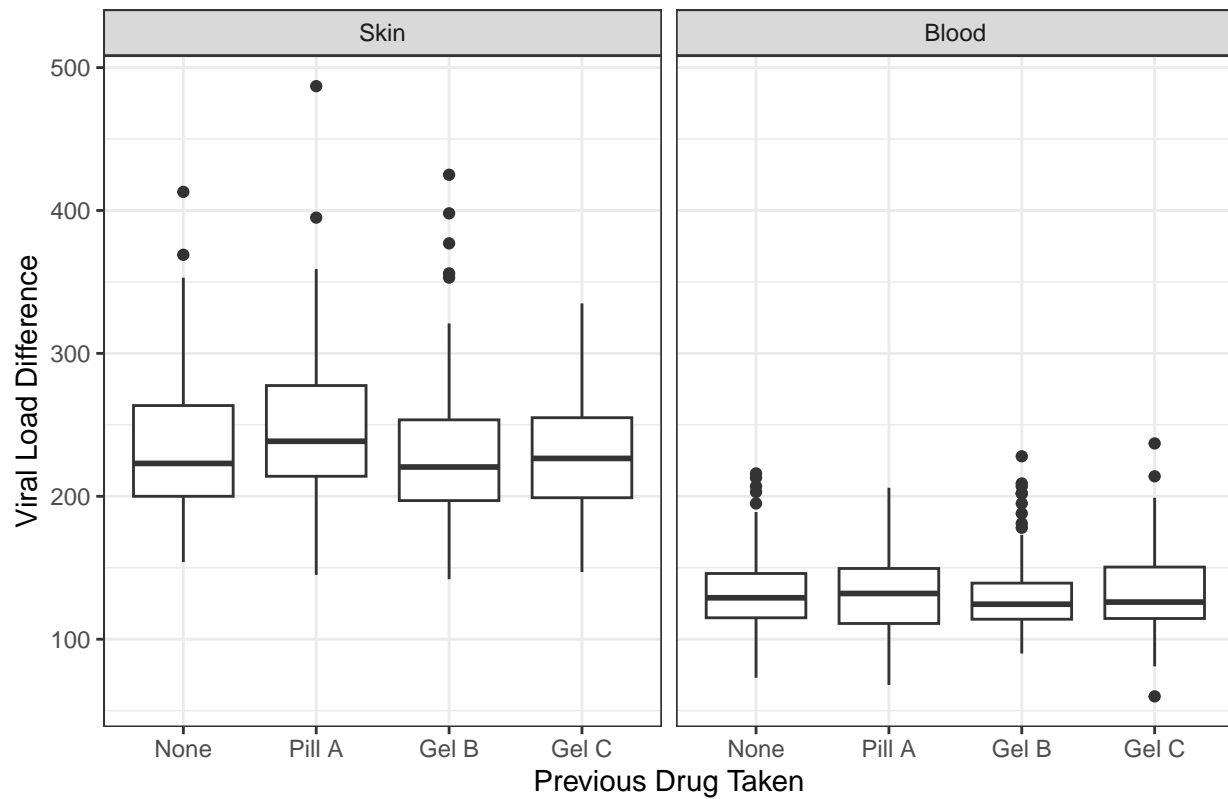
AE Record of 5 Randomly Selected Subjects



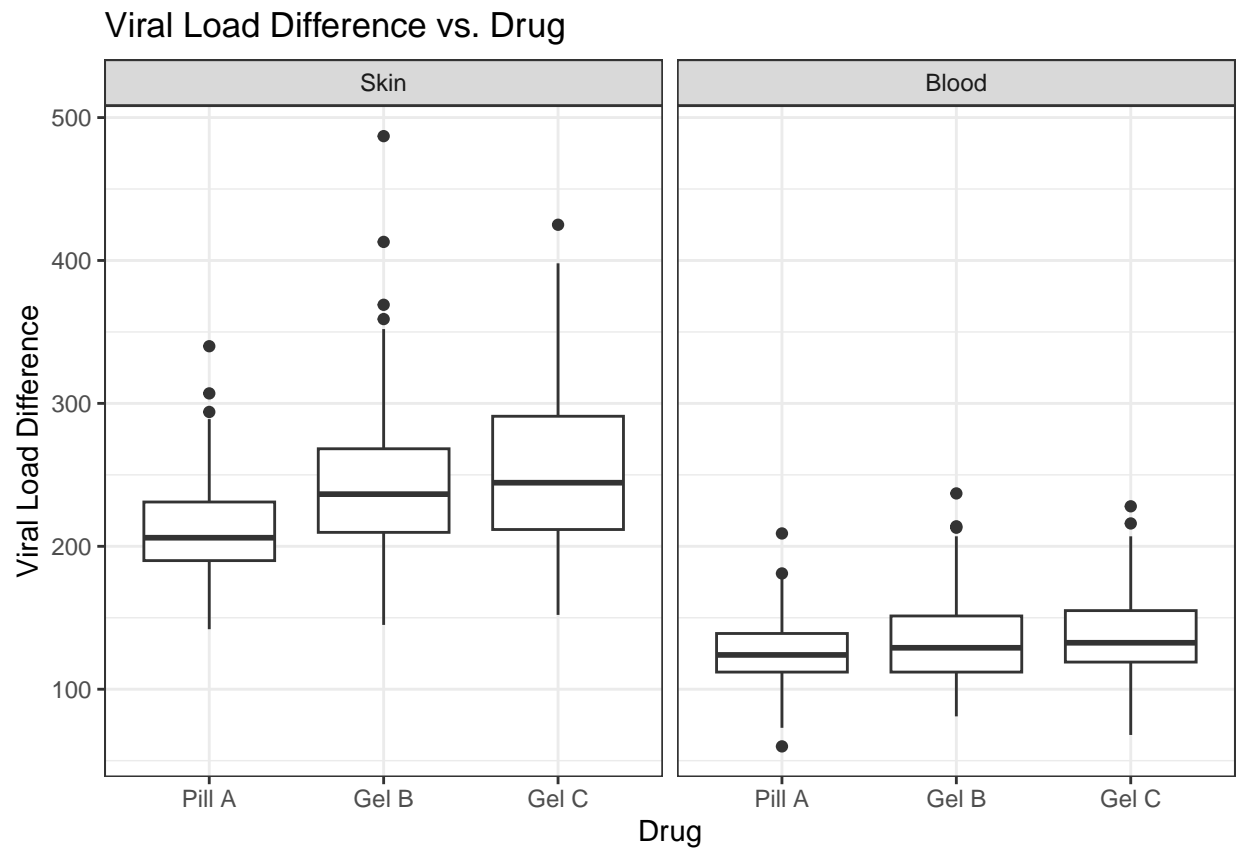
```
# endpoints.PK %>%
#   ggplot(aes(y=dvalue, x=period, color = drug)) +
#   geom_line(alpha = 0.2, aes(group = factor(ptid)))+
#   geom_point(alpha = 0.1, size = 0.8)+
#   facet_grid(measure~seq2)+
#   theme_bw()
```

```
endpoints.PK %>%
  #mutate(measure = if_else(measure == "bvial", "blood", "skin")) %>%
  ggplot(aes(y=dvalue, x=factor(drug_lag, levels=c('None', 'Pill A', 'Gel B', 'Gel C'))))+
  geom_boxplot()+
  facet_grid(.~measure)+
  xlab("Previous Drug Taken")+
  ylab("Viral Load Difference")+
  theme_bw()+
  ggtitle("Viral Load Difference vs. Previous Drug")
```

Viral Load Difference vs. Previous Drug

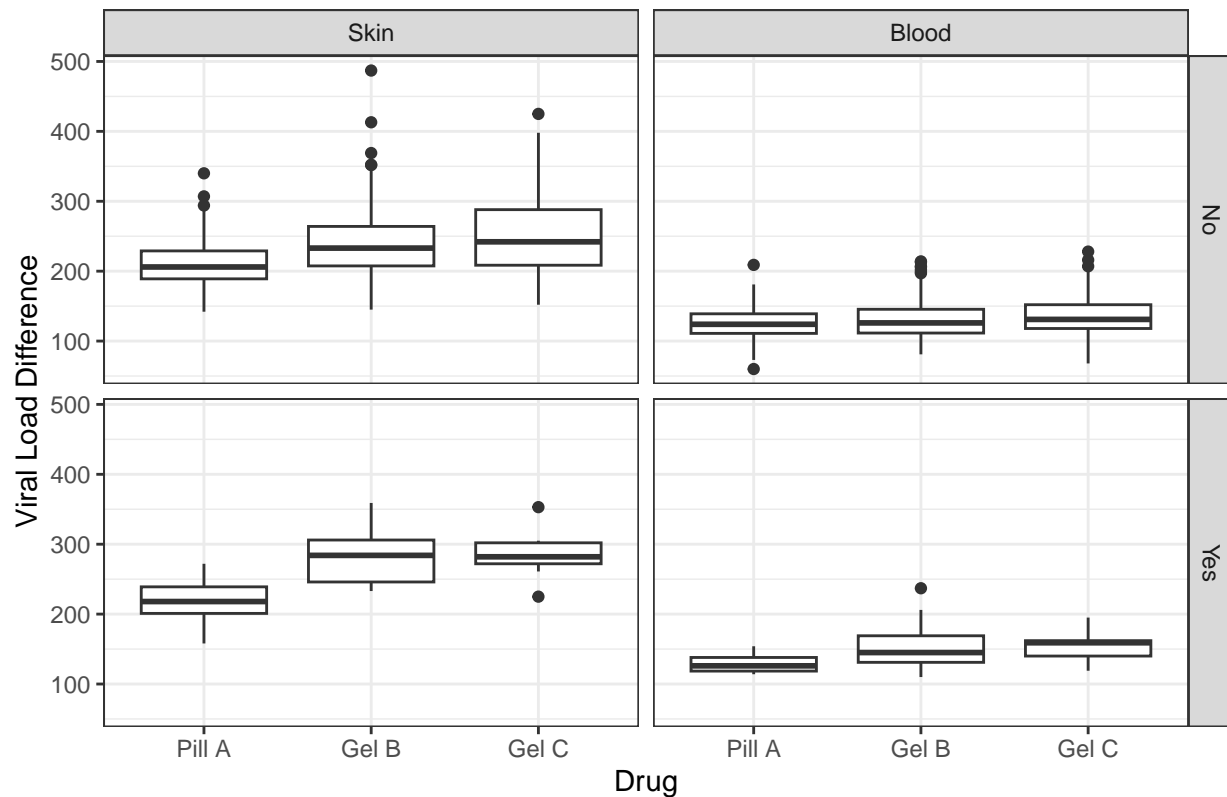


```
endpoints.PK %>%
  ggplot(aes(y=dvalue, x=factor(drug, levels=c('Pill A','Gel B','Gel C'))))+
  geom_boxplot()+
  facet_grid(.~measure)+
  xlab("Drug")+
  ylab("Viral Load Difference")+
  theme_bw()+
  ggtitle("Viral Load Difference vs. Drug")
```



```
endpoints.PK %>%
  ggplot(aes(y=dvalue, x=drug))+
  geom_boxplot()+
  facet_grid(AE_ind~measure)+
  theme_bw()+
  ylab("Viral Load Difference")+
  xlab("Drug")+
  ggtitle("Viral Load Difference vs. Adverse Event")
```

Viral Load Difference vs. Adverse Event



```
endpoints.PK %>%
  group_by(drug, measure, Adhere_sum) %>%
  summarise(dvalue = mean(dvalue)) %>%
  ggplot(aes(y=dvalue, x=Adhere_sum))+
  geom_point(size=0.8)+
  geom_smooth(method = "lm", se = FALSE, show.legend = FALSE) +
  facet_grid(drug~measure)+
  xlab("# of Adherence Days in a Treatment Period")+
  ylab("Mean Viral Load Difference")+
  ggtitle("Viral Load Difference vs. Adherence")+
  theme_bw()
```

'summarise()' has grouped output by 'drug', 'measure'. You can override using
 ## the '.groups' argument.
 ## 'geom_smooth()' using formula = 'y ~ x'

Viral Load Difference vs. Adherence

