The Impact of Music on Task Switching:

Will Lyrics Exacerbate the Switch Cost?

The main analysis for this experiment is a 2 (transition: switch vs repeat) X 3 (music condition: vocal vs instrumental vs none) within-subjects ANOVA on response times.

Update: We're also including block into the design, which is a 2 (block: early vs late) within-subjects factor.

I'll try to direct your attention to the important things you'll need for your method and results as I go. Unless I say something is important, don't pay attention to it.

In [31]:
```
library(tidyverse)
library(data.table)
library(ez)
library(xlsx)
```

In [32]:
```
current_data <- fread('../data/music_1-29.csv')
```

In [33]:
```
keep <- t(matrix(c("Subject", "subject",
                    "Procedure[SubTrial]", "procedure",
                    "CuedBlock.Sample", "block",
                    "SubTrial", "trial",
                    #"rsi","rsi",
                    "stimulus.RESP", "key_press",
                    "correct_key", "correct_key",
                    "cue", "cue",
                    "transition","transition",
                    "stimulus.RT","rt"
                     ),nrow=2))
current_data <- as.data.frame(current_data)
current_data <- current_data[,keep[,1]]
colnames(current_data) <- keep[,2]
```

In [34]:
```
current_data <- current_data %>%
  filter(procedure == 'cuedproc') %>%
  mutate(switch = ifelse(transition == 'Switch',1,0),
         error = ifelse(correct_key != key_press, 1, 0),
         taskcode = ifelse(cue == 'thumbsUpDown.jpg','valence','life'),
         #rsi = as.factor(rsi),
         subject = as.factor(subject)) %>%
  select(-one_of(c('procedure', 'cue', 'transition','key_press','correct_key'
))) %>%
  data.table()
```

In [35]:
```
head(current_data)
```

subject	block	trial	rt	switch	error	taskcode
1	1	1	9889	1	1	life
1	1	2	1439	0	0	life
1	1	3	1029	0	0	life
1	1	4	766	0	0	life
1	1	5	1952	1	0	valence
1	1	6	1891	1	0	life

In [36]:
```
subject_code <- fread('../data/subject_code.csv')[,1:2]
condition_code <- fread('../data/condition_code.csv')
```

In [37]:
```
condition_code
```

group_id	block	condition
a	1	instrumental
a	2	lyrical
a	3	control
a	4	instrumental
a	5	control
a	6	lyrical
b	1	lyrical
b	2	control
b	3	instrumental
b	4	lyrical
b	5	instrumental
b	6	control
c	1	control
c	2	instrumental
c	3	lyrical
c	4	control
c	5	lyrical
c	6	instrumental

In [38]:
```
colnames(condition_code)[1] <- 'group'
subject_code <- subject_code %>%
mutate(subject = as.factor(subject))
current_data <- current_data %>%
inner_join(subject_code) %>%
inner_join(condition_code) %>%
mutate(condition = as.factor(condition),
      transition = ifelse(switch == 1, 'switch', ifelse(switch == 0, 'repea
t', ''))) %>%
select(-group)
```

```
Joining, by = "subject"
Warning message:
"Column `subject` joining factors with different levels, coercing to characte
r vector"Joining, by = c("block", "group")
```

We're going to make sure all subjects have error rates less than 15%

- If they don't, we exclude from analysis

We're going to trim some trials based on:

- RT outliers
- Error trials and trials following error trials

In [39]:
```
bad_subjects <- current_data %>%
group_by(subject) %>%
summarize(error = mean(error)) %>%
filter(error > .15)
bad_subjects
```

subject	error
10	0.5021368
11	0.1944444
12	0.2200855
13	0.5149573
14	0.1581197
16	0.1816239
20	0.7200855
21	0.3461538
29	0.5256410
3	0.1965812
4	0.1623932
5	0.5106838
7	0.5299145
8	0.1709402

LOL, that's not good. Error rates this high are pretty uncommon. Let's stop to do a descriptive analysis on these error rates. I'm going to exclude everyone above 25%, because that's just ridiculous. Let's see if we can figure out why / when so many people are making errors.

Impromptu Error Analysis

Filter out the subjects with > 25% error rates.

In [40]:
```
bad_subjects <- current_data %>%
group_by(subject) %>%
summarize(error = mean(error)) %>%
filter(error > .25)
bad_subjects

N <- current_data %>%
group_by(subject) %>%
summarize(n()) %>%
nrow()
paste0('Total N is ',N,', number of subjects above error threshold is ', nrow(
bad_subjects), ', ',
       N - nrow(bad_subjects), ' subjects retained for analysis.')
```

subject	error
10	0.5021368
13	0.5149573
20	0.7200855
21	0.3461538
29	0.5256410
5	0.5106838
7	0.5299145

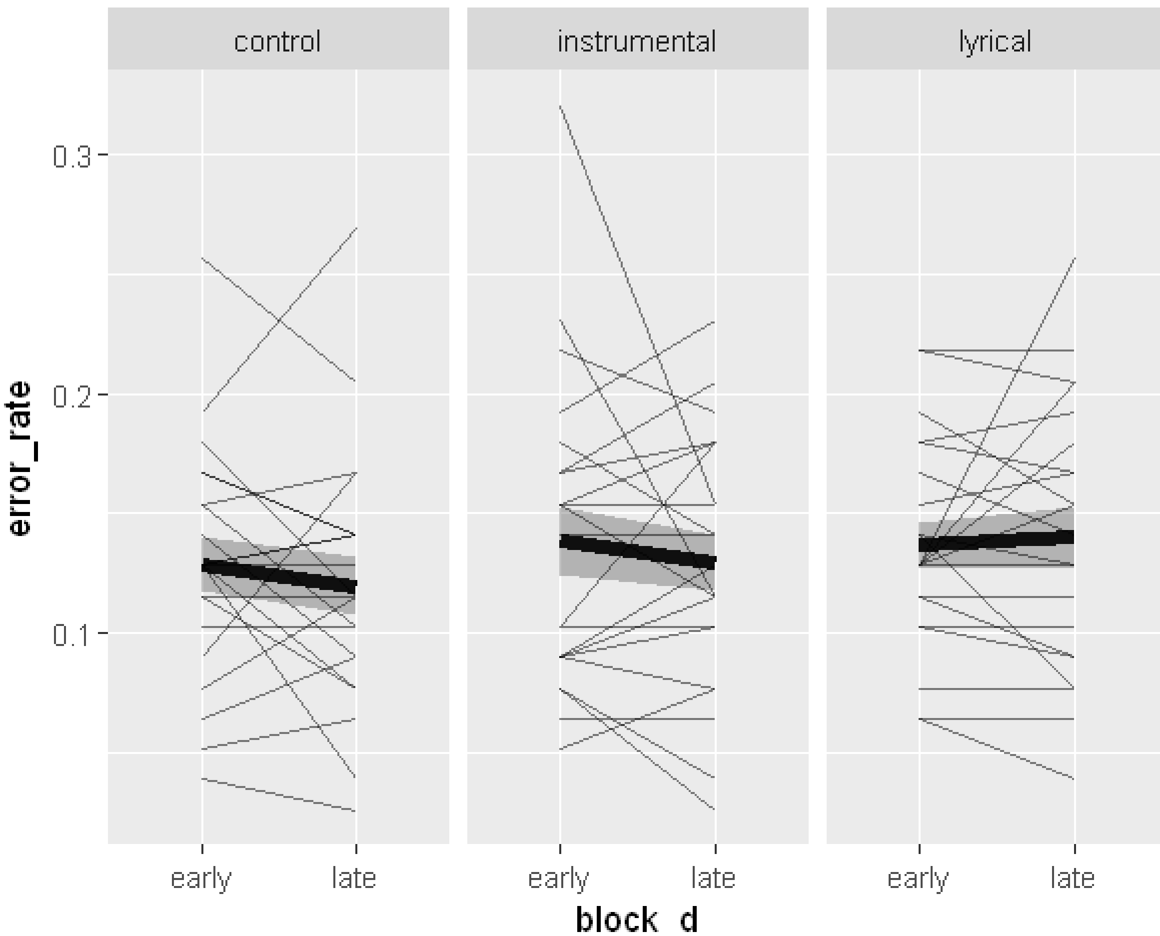
'Total N is 28, number of subjects above error threshold is 7, 21 subjects retained for analysis.'

You'll want to report the error threshold (25%) and how many subjects were excluded because of it (7).

In [41]:
```
## getting rid of error subjects
current_data <- current_data[!(current_data$subject %in% bad_subjects$subject
),]
```

It's possible that subjects are either getting better with practice (making fewer errors as they go), or they're getting tired (making more errors as they go). We'll plot error rate as a function of subject, condition, and block.

In [42]:
```
options(repr.plot.width = 5, repr.plot.height = 4)
subject_means <- current_data %>%
mutate(block_d = factor(ifelse(block < 4, 'early', 'late'))) %>%
group_by(subject, block_d, condition) %>%
summarize(error = mean(error))

subject_means %>%
group_by(block_d, condition) %>%
summarize(error_rate = mean(error), se = sd(error) / sqrt(n())) %>%
ggplot(aes(x = block_d, y = error_rate, group = 1)) +
geom_line(size = 2) +
geom_ribbon(aes(ymin = error_rate - se, ymax = error_rate + se), alpha = .3) +
geom_line(data = subject_means, aes(x = block_d, y = error, group = subject),
 alpha = .5) +
facet_wrap(~condition) +
theme(legend.position = 'none')
```



Error rates seem to decrease slightly across blocks for the control and instrumental conditions, but for lyrical error rates increase slightly or remain the same. My sense is that some people just got the keys wrong at some points.

Below, I'm outputting the mean error rates broken down by:

- The original design
- The original design plus block

In [43]:
```
current_data <- current_data %>%
mutate(block_d = factor(ifelse(block < 4, 'early', 'late')))

current_data %>%
group_by(subject, condition, transition) %>%
summarize(error = mean(error)) %>%
group_by(condition, transition) %>%
summarize('Error Rate (Mean)' = mean(error), 'Error Rate (SE)' = sd(error) / s
qrt(n()))

current_data %>%
group_by(subject, condition, transition, block_d) %>%
summarize(error = mean(error)) %>%
group_by(condition, transition, block_d) %>%
summarize('Error Rate (Mean)' = mean(error), 'Error Rate (SE)' = sd(error) / s
qrt(n()))
```

condition	transition	Error Rate (Mean)	Error Rate (SE)
control	repeat	0.1034093	0.009612008
control	switch	0.1441583	0.013407791
instrumental	repeat	0.1174777	0.011962529
instrumental	switch	0.1502676	0.016974272
lyrical	repeat	0.1217577	0.010434840
lyrical	switch	0.1547279	0.013814117

condition	transition	block_d	Error Rate (Mean)	Error Rate (SE)
control	repeat	early	0.1063509	0.01282452
control	repeat	late	0.1004741	0.01310627
control	switch	early	0.1500850	0.01450746
control	switch	late	0.1381907	0.01765882
instrumental	repeat	early	0.1163953	0.01317018
instrumental	repeat	late	0.1197273	0.01436433
instrumental	switch	early	0.1589485	0.02054473
instrumental	switch	late	0.1418177	0.01653885
lyrical	repeat	early	0.1237938	0.01134526
lyrical	repeat	late	0.1207749	0.01304317
lyrical	switch	early	0.1506573	0.01454157
lyrical	switch	late	0.1597965	0.01627622

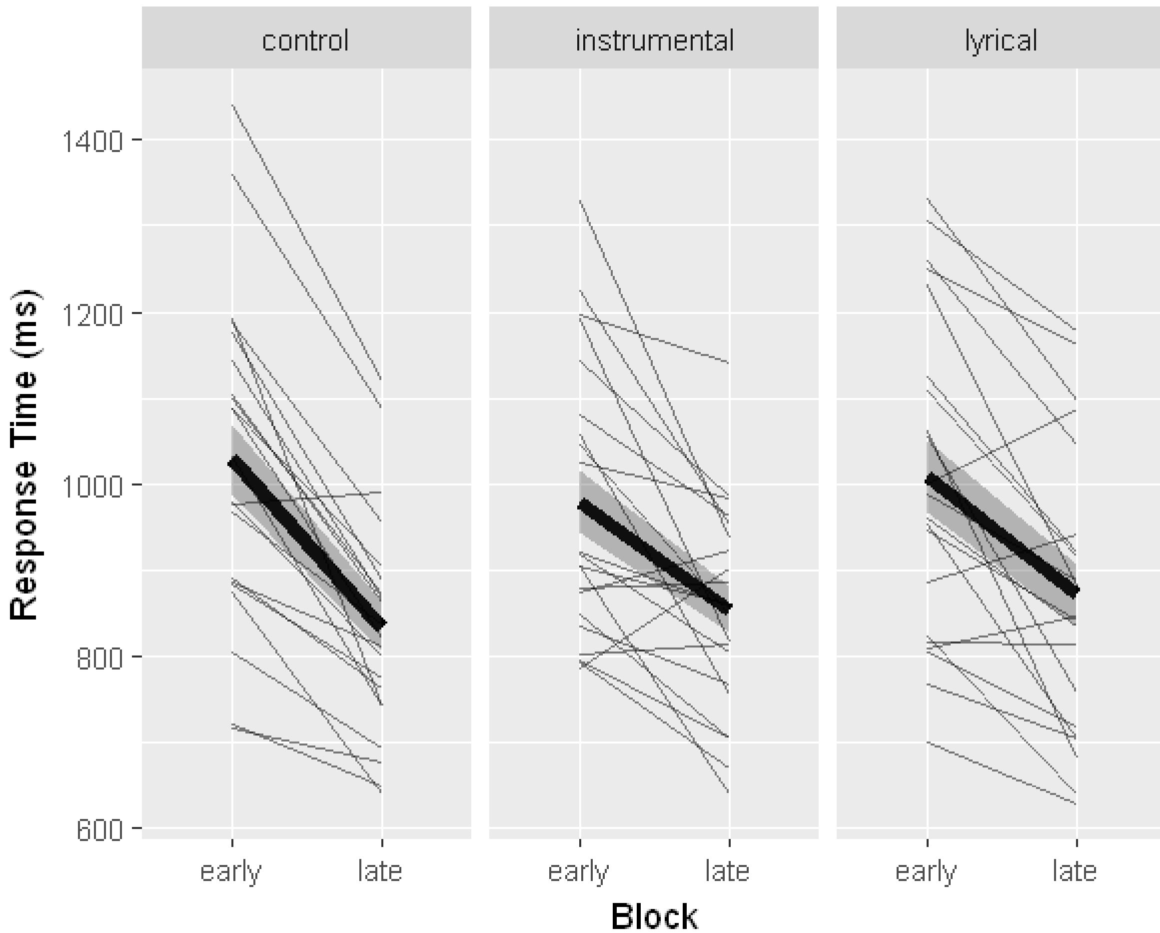
Let's look at the data broken down the same way but for RTs (after removing error trials and RT outliers). I'll output all the means for RT near the end of the documnet, closer to the analysis.

In [44]:
```
before_trimming <- nrow(current_data)

rt_stats <- current_data %>%
group_by(subject) %>%
summarize(rt_mean = mean(rt), rt_sd = sd(rt))

current_data <- current_data %>%
inner_join(rt_stats) %>%
filter(error == 0 & shift(error) == 0,
      rt < rt_mean + 2 * rt_sd & rt > rt_mean - 2 * rt_sd) %>%
select(-rt_mean, -rt_sd)

after_trimming <- nrow(current_data)
paste0('Trimming trials according to errors and response times resulted in the
 loss of ',
       round((1 - (after_trimming / before_trimming)) * 100, 2),
      '% of the data')
```

```
Joining, by = "subject"
```

'Trimming trials according to errors and response times resulted in the loss of 27.58% of the data'

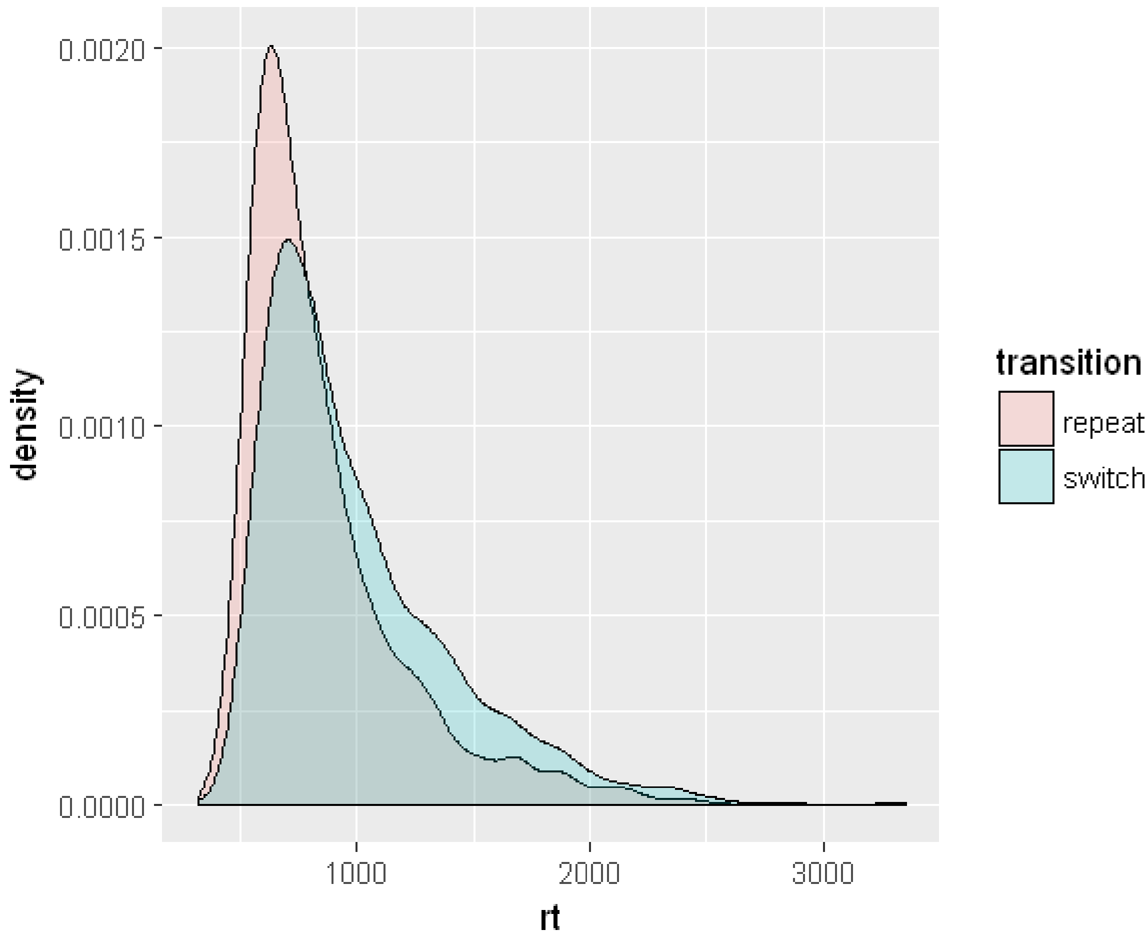
In [45]:
```
subject_means <- current_data %>%
mutate(block_d = factor(ifelse(block < 4, 'early', 'late'))) %>%
group_by(subject, block_d, condition) %>%
summarize(rt = mean(rt))

subject_means %>%
group_by(block_d, condition) %>%
summarize(r_time = mean(rt), se = sd(rt) / sqrt(n())) %>%

ggplot(aes(x = block_d, y = r_time, group = 1)) +
geom_line(size = 2) +
geom_ribbon(aes(ymin = r_time - se, ymax = r_time + se), alpha = .3) +
geom_line(data = subject_means, aes(x = block_d, y = rt, group = subject), alp
ha = .5) +
facet_wrap(~condition) +
theme(legend.position = 'none') + ylab('Response Time (ms)') + xlab('Block')
```



Unlike error rates, RTs decrease dramatically over blocks, seemingly regardless of condition.

Let's start to look for the switch cost. We know RTs on switch trials should be longer than that for repeat trials:

In [46]:
```
ggplot(current_data, aes(x = rt)) + geom_density(aes(fill = transition), alpha
 = .2)
```



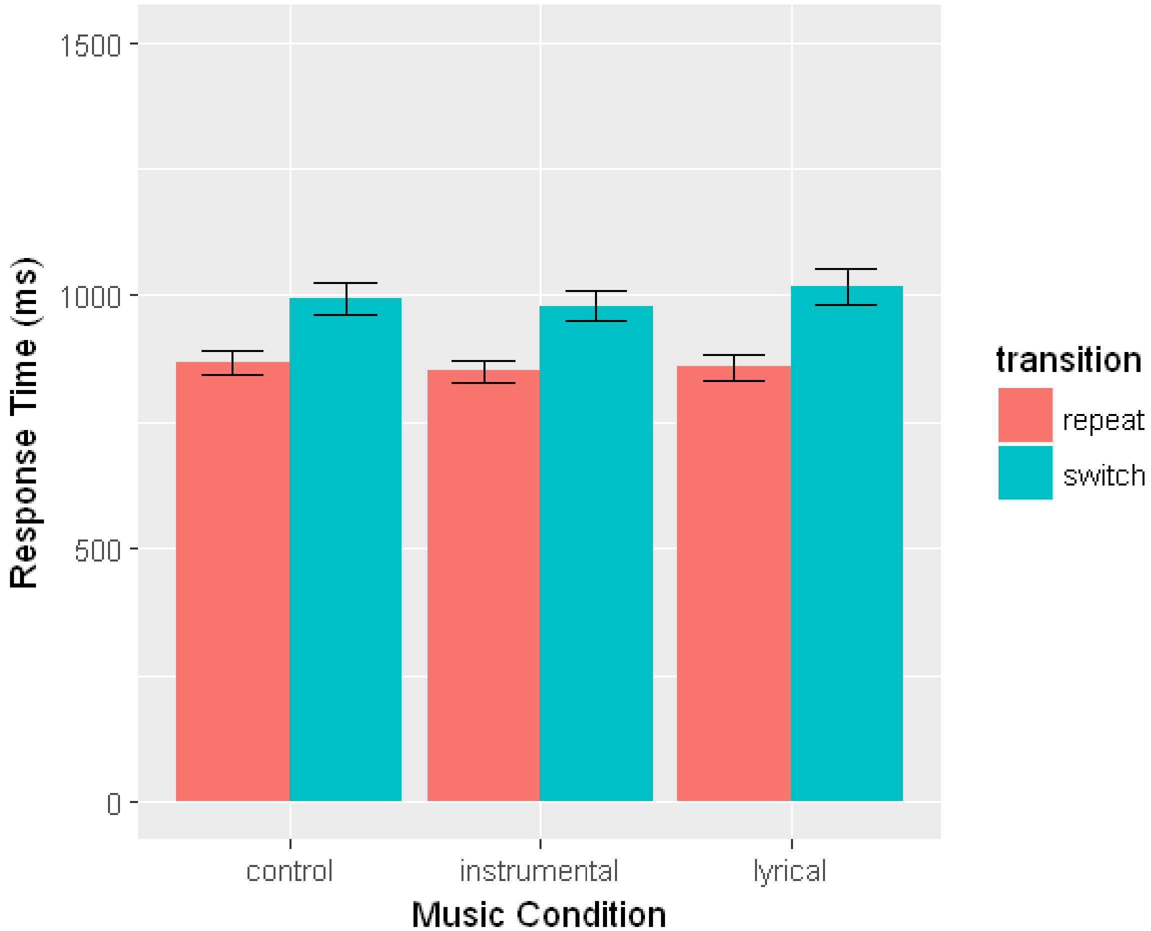
Plotting the Original Design

We'll plot the 2 (transition: repeat vs switch) X 2 (music condition: control vs lyrical vs instrumental) design predicting RTs.

In [47]:
```
head(current_data)
```

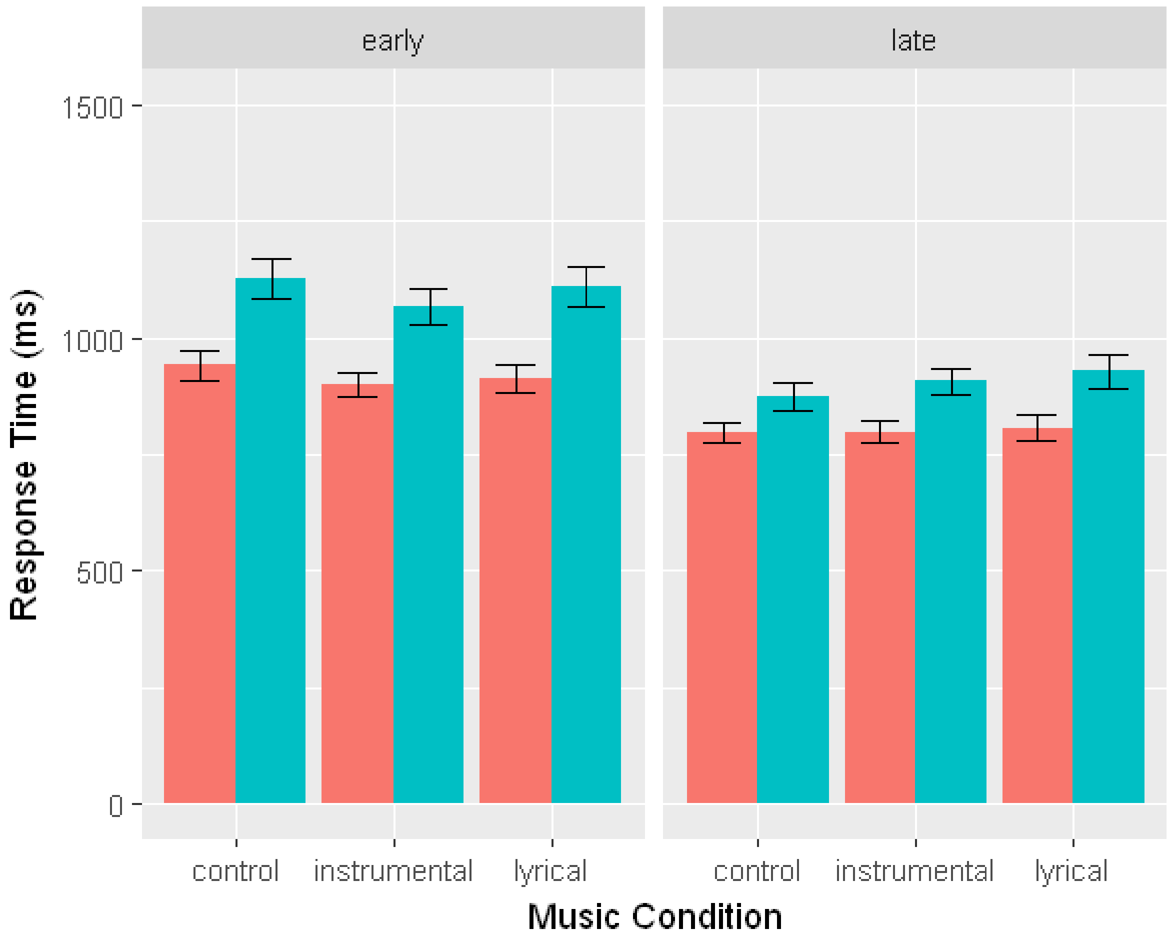
subject	block	trial	rt	switch	error	taskcode	condition	transition	block_d
1	1	3	1029	0	0	life	instrumental	repeat	early
1	1	4	766	0	0	life	instrumental	repeat	early
1	1	5	1952	1	0	valence	instrumental	switch	early
1	1	6	1891	1	0	life	instrumental	switch	early
1	1	7	902	0	0	life	instrumental	repeat	early
1	1	8	2131	1	0	valence	instrumental	switch	early

In [48]:
```
current_data %>%
group_by(subject, condition, transition) %>%
summarize(rt = mean(rt)) %>%
group_by(condition, transition) %>%
summarize(r_time = mean(rt), se = sd(rt) / sqrt(N)) %>%
ggplot(aes(x = condition, y = r_time, group = transition)) +
geom_bar(stat = 'identity', aes(fill = transition), position = 'dodge') +
geom_errorbar(stat = 'identity', aes(ymin = r_time - se, ymax = r_time + se),
 position = position_dodge(width = .9), width = .5) +
xlab('Music Condition') + ylab('Response Time (ms)') + ylim(0, 1500)
```



Well there's definitely a switch cost. Does the switch cost vary across musical condition... *maybe* a little more exaggerated in the lyrical condition? I'll break this down by block:

In [49]:
```
current_data %>%
group_by(subject, condition, transition, block_d) %>%
summarize(rt = mean(rt)) %>%
group_by(condition, transition, block_d) %>%
summarize(r_time = mean(rt), se = sd(rt) / sqrt(N)) %>%
ggplot(aes(x = condition, y = r_time, group = transition)) +
geom_bar(stat = 'identity', aes(fill = transition), position = 'dodge') +
geom_errorbar(stat = 'identity', aes(ymin = r_time - se, ymax = r_time + se),
 position = position_dodge(width = .9), width = .5) +
xlab('Music Condition') + ylab('Response Time (ms)') + ylim(0, 1500) + facet_g
rid(~block_d) + theme(legend.position = 'none')
```



Here, it's easy to see how overall RTs and switch costs decrease over blocks.

The ANOVA

Below is the ANOVA for the original design (without block).

In [50]:
```
m1 <- ezANOVA(data = current_data, wid = subject, within = .(condition, transi
tion), dv = rt, detailed = TRUE)
cbind(m1$ANOVA, n2p = m1$ANOVA$SSn / (m1$ANOVA$SSn + m1$ANOVA$SSd))
m1[2:3]
```

```
Warning message:
"Converting "subject" to factor for ANOVA."Warning message:
"Converting "transition" to factor for ANOVA."Warning message:
"Collapsing data to cell means. *IF* the requested effects are a subset of th
e full design, you must use the "within_full" argument, else results may be i
naccurate."
```

Effect	DFn	DFd	SSn	SSd	F	p	p<.05
(Intercept)	1	20	1.084354e+08	2181466.70	994.1510643	1.581980e-18	*
condition	2	40	1.177256e+04	456802.31	0.5154337	6.011542e-01	
transition	1	20	6.003806e+05	84978.27	141.3021579	1.606526e-10	*
condition:transition	2	40	6.810522e+03	96661.74	1.4091453	2.562202e-01	

$`Mauchly's Test for Sphericity`

	Effect	W	p	p<.05
2	condition	0.8699656	0.2662378	
4	condition:transition	0.8934088	0.3427496	

$`Sphericity Corrections`

	Effect	GGe	p[GG]	p[GG]<.05	HFe	p[HF]	p[HF]<.05
2	condition	0.8849288	0.5796325		0.9645292	0.5947836	
4	condition:transition	0.9036761	0.2566924		0.9881573	0.2563035	

So you'll obviously use the values above if you wanted to report effects from the original design.

An ANOVA looking at the original design plus block:

In [51]:
```
m2 <- ezANOVA(data = current_data, wid = subject, within = .(condition, transition, block_d), dv = rt, detailed = TRUE)
cbind(m2$ANOVA, n2p = m2$ANOVA$SSn / (m2$ANOVA$SSn + m2$ANOVA$SSd))
m2[2:3]
```

```
Warning message:
"Converting "subject" to factor for ANOVA."Warning message:
"Converting "transition" to factor for ANOVA."Warning message:
"Collapsing data to cell means. *IF* the requested effects are a subset of th
e full design, you must use the "within_full" argument, else results may be i
naccurate."
```

Effect	DFn	DFd	SSn	SSd	F	p
(Intercept)	1	20	2.182089e+08	4366607.51	999.4437386	1.501669e-18
condition	2	40	2.134099e+04	1007548.77	0.4236220	6.575740e-01
transition	1	20	1.284503e+06	162413.25	158.1771628	5.904292e-11
block_d	1	20	1.570063e+06	317524.71	98.8938831	3.473648e-09
condition:transition	2	40	8.755189e+03	196312.41	0.8919649	4.178440e-01
condition:block_d	2	40	5.340709e+04	623942.22	1.7119242	1.934789e-01
transition:block_d	1	20	1.045908e+05	85638.26	24.4261878	7.849060e-05
condition:transition:block_d	2	40	6.185649e+03	227214.66	0.5444762	5.843842e-01

$`Mauchly's Test for Sphericity`

	Effect	W	p	p<.05
2	condition	0.8830223	0.3067120	
5	condition:transition	0.9326936	0.5158465	
6	condition:block_d	0.9952976	0.9562092	
8	condition:transition:block_d	0.8476806	0.2080686	

$`Sphericity Corrections`

	Effect	GGe	p[GG]	p[GG]<.05	HFe	p[HF]	p[HF]<.05
2	condition	0.8952730	0.6358188		0.9775545	0.6531098	
5	condition:transition	0.9369381	0.4123711		1.0303195	0.4178440	
6	condition:block_d	0.9953196	0.1936510		1.1050759	0.1934789	
8	condition:transition:block_d	0.8678150	0.5604355		0.9430446	0.5744572	

And if you want to report the analysis with block as a factor, use the estimates from above.

Following up on the Block X Transition Interaction

Whenever you get an interaction, it's good to follow up on it to determine exactly what is going on. I'm going to do that below and try to be very clear about what you need to know.

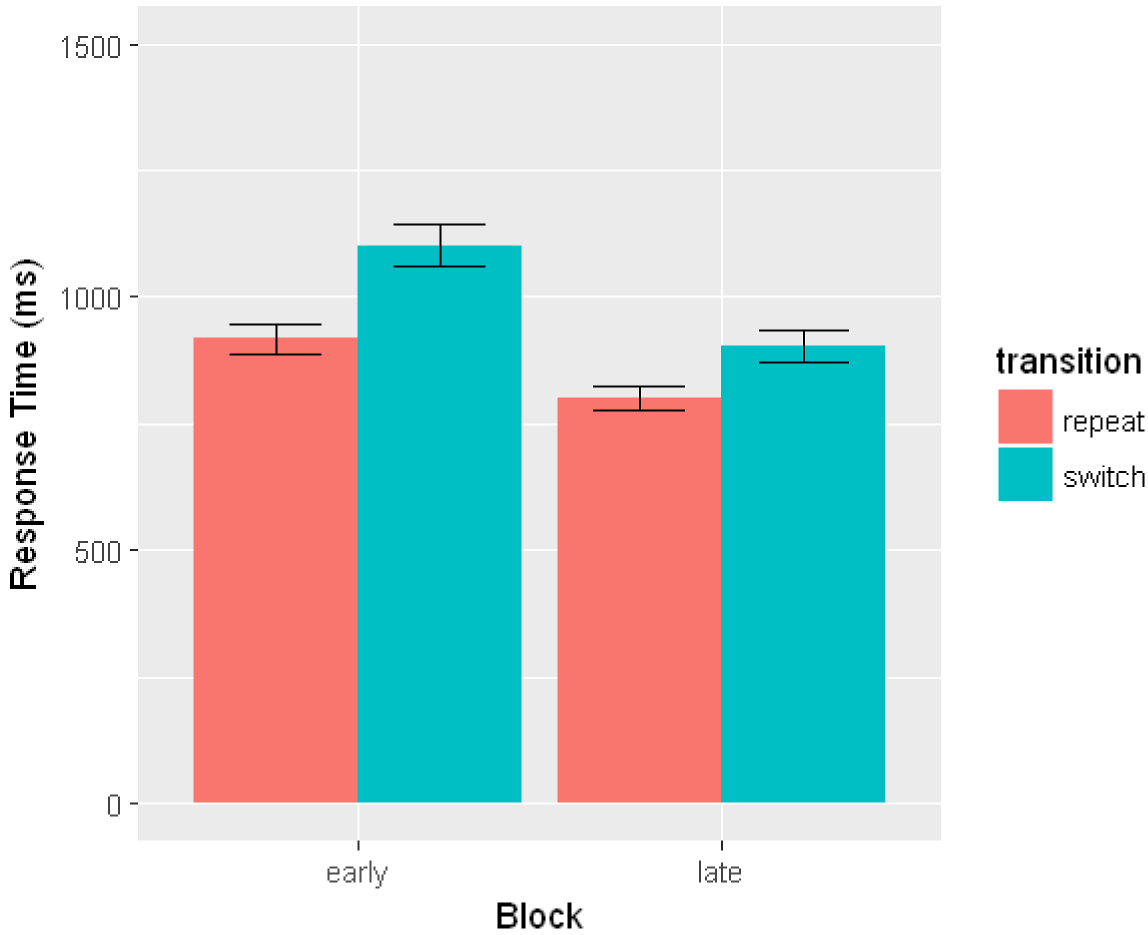
Here is my order that I'll present the follow-up information in:

- Plot the 2 X 2 graph
- Output the relevant means

You'll use the last bullet point to describe the interaction in text and make your own graphs.

Plots

In [52]:
```
current_data %>%
group_by(subject, condition, transition,block_d) %>%
summarize(rt = mean(rt)) %>%
group_by(block_d, transition) %>%
summarize(r_time = mean(rt), se = sd(rt) / sqrt(N)) %>%
ggplot(aes(x = block_d, y = r_time, group = transition)) +
geom_bar(stat = 'identity', aes(fill = transition), position = 'dodge') +
geom_errorbar(stat = 'identity', aes(ymin = r_time - se, ymax = r_time + se),
 position = position_dodge(width = .9), width = .5) +
xlab('Block') + ylab('Response Time (ms)') + ylim(0, 1500)
```



So we can pretty clearly see that--while overall RTs are lower in late relative to early blocks--the switch cost also decreases over blocks.

How do we describe this?

The switch cost decreases as people progress through the experiment. There is still a switch cost in later portions of the experiment, but it is not as strong as in earlier portions.

Follow-up Tests

EDIT. Note to Pat: So I'll delete all this? Want to confirm first bc it's a lot of code.

In [53]:
```
current_data <- as.data.table(current_data)
SSd <- m2$ANOVA$SSd[7]
DFd <- m2$ANOVA$DFd[7]

early_block <- ezANOVA(data = current_data[block_d == 'early'],
                       wid = subject, within = .(transition), dv = rt, detai
led = TRUE)

SSn <- early_block$ANOVA$SSn[2]
DFn <- early_block$ANOVA$DFn[2]
MSn <- SSn / DFn
MSe <- SSd / DFd
F <- MSn / MSe
p <- pf(F, DFn, DFd, lower.tail = FALSE)
n2p <- SSn / (SSn + SSd)
paste0('Simple effect of transition for early blocks:')
paste0('F(', DFn, ', ', DFd,') = ', round(F,2), ', p = ', round(p, 4), ', n2
p = ', round(n2p, 2))
```

```
Warning message:
"Converting "subject" to factor for ANOVA."Warning message:
"Converting "transition" to factor for ANOVA."Warning message:
"Collapsing data to cell means. *IF* the requested effects are a subset of th
e full design, you must use the "within_full" argument, else results may be i
naccurate."
```

'Simple effect of transition for early blocks:'

'F(1, 20) = 83.39, p = 0, n2p = 0.81'

Remember to write *p* as: *p* < .001

In [54]:
```
current_data <- as.data.table(current_data)
SSd <- m2$ANOVA$SSd[7]
DFd <- m2$ANOVA$DFd[7]

early_block <- ezANOVA(data = current_data[block_d == 'late'],
                       wid = subject, within = .(transition), dv = rt, detaile
d = TRUE)

SSn <- early_block$ANOVA$SSn[2]
DFn <- early_block$ANOVA$DFn[2]
MSn <- SSn / DFn
MSe <- SSd / DFd
F <- MSn / MSe
p <- pf(F, DFn, DFd, lower.tail = FALSE)
n2p <- SSn / (SSn + SSd)
paste0('Simple effect of transition for late blocks:')
paste0('F(', DFn, ', ', DFd,') = ', round(F,2), ', p = ', round(p, 4), ', n2p
 = ', round(n2p, 2))
```

```
Warning message:
"Converting "subject" to factor for ANOVA."Warning message:
"Converting "transition" to factor for ANOVA."Warning message:
"Collapsing data to cell means. *IF* the requested effects are a subset of th
e full design, you must use the "within_full" argument, else results may be i
naccurate."
```

'Simple effect of transition for late blocks:'

'F(1, 20) = 24.52, p = 1e-04, n2p = 0.55'

Like I thought, the simple effect of transition is significant for both early and late blocks, but the effect is stronger (as reflected by the partial eta squared, or n2p) for early rather than late blocks. You'll want to report these F tests in text when you describe the follow up tests to the interaction.

Means and Standard Errors

Here are the means and standard errors for the full design (with and without block), and for the effects that reached significance.

Descriptives for the full designs:

In [55]:
```
cell_means <- current_data %>%
group_by(subject, condition, transition, block_d) %>%
summarize(rt = mean(rt)) %>%
group_by(condition, transition, block_d) %>%
summarize('Response Time (Mean)' = mean(rt), 'Response Time (SE)' = sd(rt) / s
qrt(n()))

cell_means

cell_means %>%
group_by(transition, condition) %>%
summarize('Response Time (Mean )' = mean(`Response Time (Mean)`), 'Response Ti
me (SE)' = sd(`Response Time (Mean)`) / sqrt(n()))
```

condition	transition	block_d	Response Time (Mean)	Response Time (SE)
control	repeat	early	941.3095	36.20716
control	repeat	late	796.9394	26.47842
control	switch	early	1125.8575	48.15168
control	switch	late	873.5179	32.65068
instrumental	repeat	early	898.7964	28.23566
instrumental	repeat	late	797.8107	26.43389
instrumental	switch	early	1068.0331	45.16564
instrumental	switch	late	906.7497	31.82994
lyrical	repeat	early	913.0164	36.23228
lyrical	repeat	late	807.0102	31.57730
lyrical	switch	early	1109.8368	49.90954
lyrical	switch	late	927.6267	43.46871

transition	condition	Response Time (Mean)	Response Time (SE)
repeat	control	869.1244	72.18504
repeat	instrumental	848.3036	50.49285
repeat	lyrical	860.0133	53.00309
switch	control	999.6877	126.16981
switch	instrumental	987.3914	80.64173
switch	lyrical	1018.7317	91.10506

Descriptives for the significant effects:

In [56]:
```
cell_means %>%
group_by(transition) %>%
summarize('Response Time (Mean )' = mean(`Response Time (Mean)`), 'Response Ti
me (SE)' = sd(`Response Time (Mean)`) / sqrt(n()))

cell_means %>%
group_by(transition, block_d) %>%
summarize('Response Time (Mean )' = mean(`Response Time (Mean)`), 'Response Ti
me (SE)' = sd(`Response Time (Mean)`) / sqrt(n()))
```

transition	Response Time (Mean)	Response Time (SE)
repeat	859.1471	26.81718
switch	1001.9369	45.62217

transition	block_d	Response Time (Mean)	Response Time (SE)
repeat	early	917.7074	12.494591
repeat	late	800.5868	3.221545
switch	early	1101.2425	17.236705
switch	late	902.6314	15.754988

Let me know if you have any questions about how to interpret all of this.

Survey Data

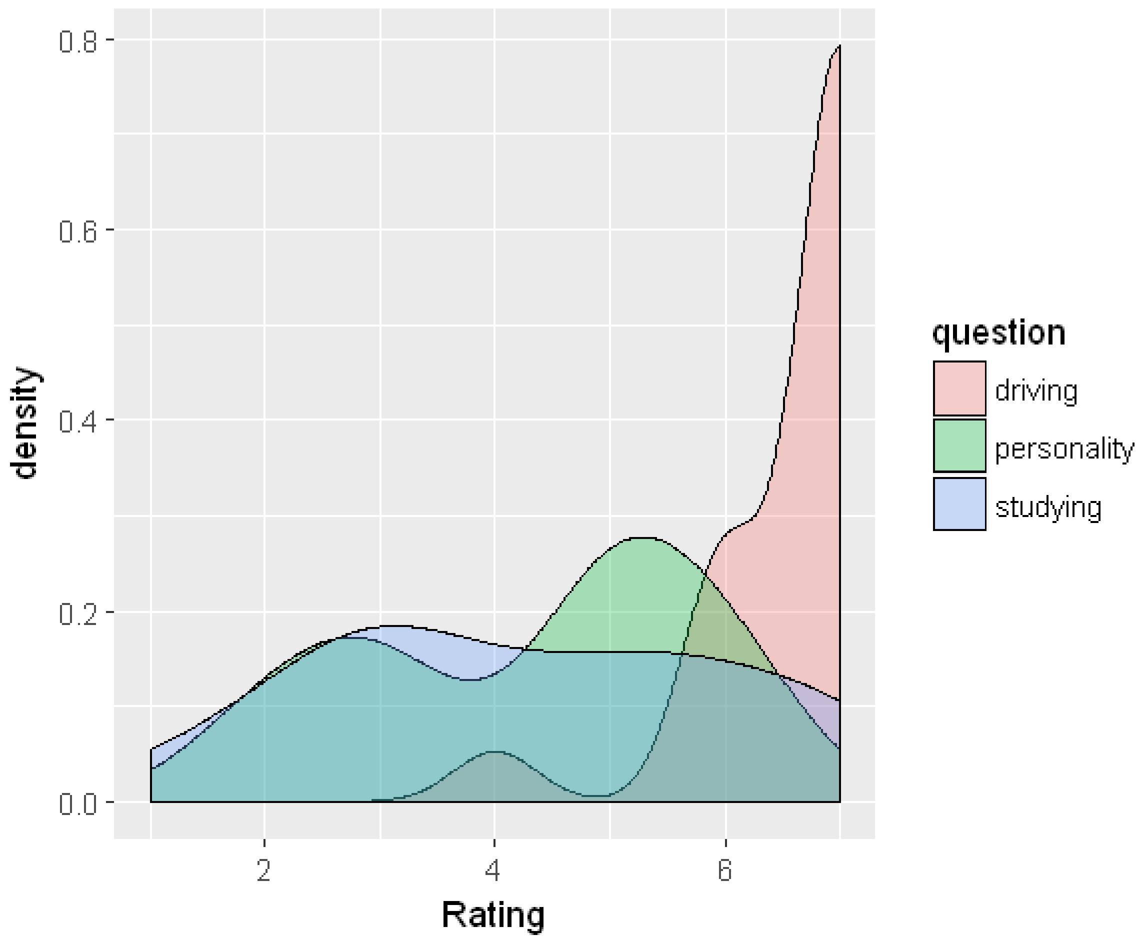
Doing a quick descriptive summary of the survey data. I'm dropping from these data those subjects that we dropped for the main analysis.

In [57]:
```
survey <- fread('../data/survey_results.csv')
survey <- survey[complete.cases(survey)]
survey <- survey[!(subject %in% bad_subjects$subject)]
head(survey)
```

subject	personality	driving	studying
1	6	7	3
2	5	7	4
3	5	7	2
4	6	7	6
6	3	6	5
8	5	7	7

In [58]:
```
survey %>%
gather(question, rating, personality:studying) %>%
ggplot(aes(x = rating)) + geom_density(aes(fill = question), alpha = .3) + x
lab('Rating')

survey %>%
gather(question, rating, personality:studying) %>%
group_by(question) %>%
summarize('Rating (Mean)' = mean(rating), 'Rating (SD)' = sd(rating))
```

question	Rating (Mean)	Rating (SD)
driving	6.619048	0.7400129
personality	4.333333	1.4605935
studying	4.285714	1.7928429



Scores tended to be pretty uniform for both personality and studying, but most people seem to listen to music while driving.

Data Access

If you guys want to access the data to do stuff to it yourselves, everything is in the `data` folder inside your guys' main folder on OSF. More specfically: `./Music Group/data/`

- `survey_results.xlsx` -- the survey data that's plotted immediately above
- `subject_means.xlsx` -- a cleaned dataset with the full design by subject in excel

In [59]:
```
library(reshape2)
current_data %>%
group_by(subject, block_d, condition, transition) %>%
summarize(rt = mean(rt)) %>%
dcast(subject ~ block_d + condition + transition, value.var = 'rt') %>%
mutate(subject = as.numeric(subject)) %>%
arrange(subject) %>%
write.xlsx('../data/subject_means.xlsx', row.names = FALSE)
```