Data Analysis

I cleaned the data behind the scenes, this notebook will be for running and interpreting the analyses.
The first portion of this is just importing data and stuff.. I'll let you know when it's time to pay attention.

In [33]:
```
library(tidyverse)
library(data.table)
library(ez)
library(xlsx)
```

```
Loading required package: rJava
Loading required package: xlsxjars
```

In [2]:
```
current_data <- fread('../data/integrity_tidy.csv', stringsAsFactors = TRUE)
head(current_data)
str(current_data)
```

subject	difficulty	curve	question_number	action	frustrate
1	diff	com	1	2	3
1	diff	com	2	3	NA
1	diff	com	3	3	NA
1	diff	com	4	3	NA
1	diff	non	1	3	2
1	diff	non	2	3	NA

```
Classes 'data.table' and 'data.frame':  944 obs. of  6 variables:
 $ subject        : int  1 1 1 1 1 1 1 1 1 1 ...
 $ difficulty     : Factor w/ 2 levels "diff","easy": 1 1 1 1 1 1 1 1 2 2 ...
 $ curve          : Factor w/ 2 levels "com","non": 1 1 1 1 2 2 2 2 1 1 ...
 $ question_number: Factor w/ 4 levels "1","2","3","4": 1 2 3 4 1 2 3 4 1 2
...
 $ action         : int  2 3 3 3 3 3 3 3 2 3 ...
 $ frustrate      : int  3 NA NA NA 2 NA NA NA 6 NA ...
 - attr(*, ".internal.selfref")=<externalptr>
```

Outline

Let's handle the frustration data first, because it's simpler. So the order will be:

- Frustration Analysis
  - Plot
  - ANOVA
  - Means
- Action Analysis
  - Plots
    - Broken down by question
  - ANOVAs
    - We need five total:
      - One overall just to get the main effects
      - One for each question
    - Any follow-up analyses
  - Means for everything

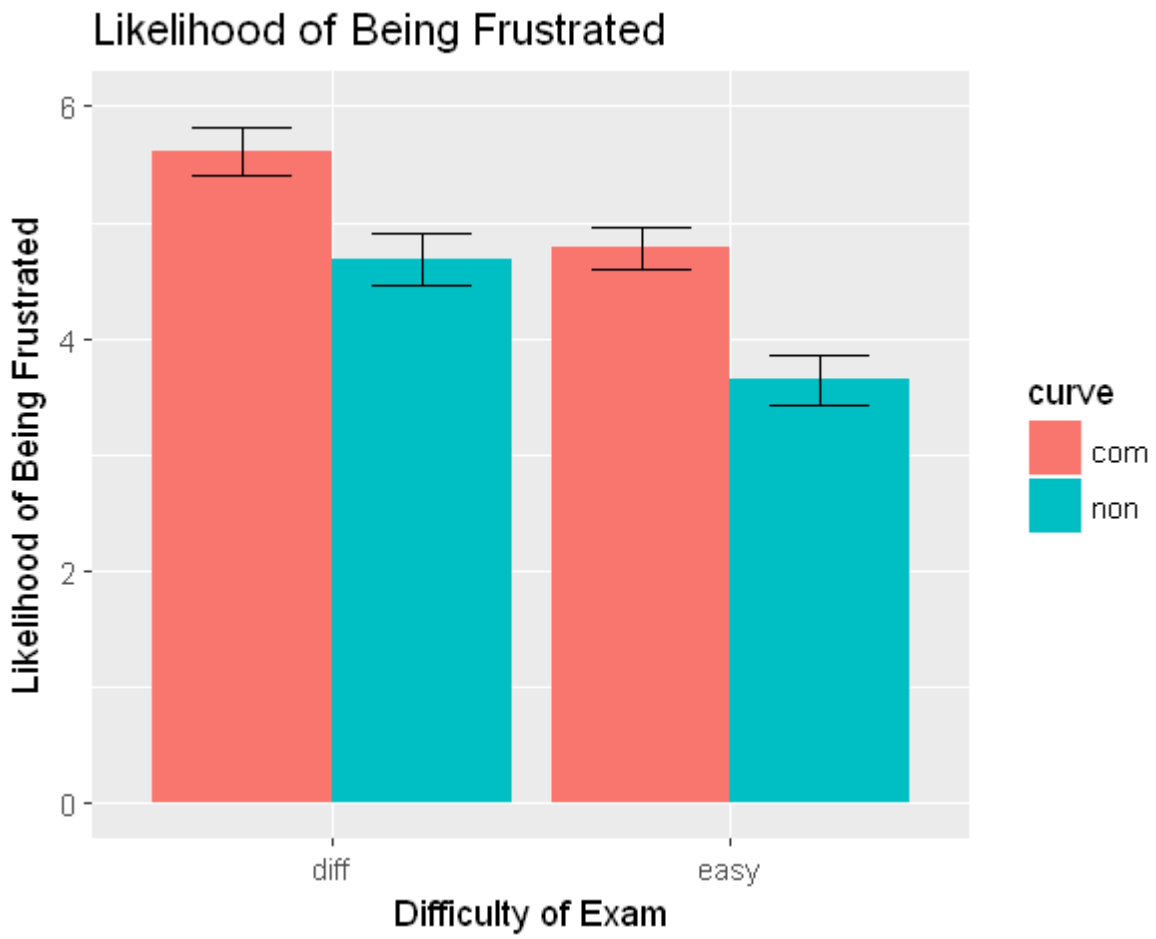
Frustration Analysis

Plot

In [6]:
```
options(repr.plot.width = 5, repr.plot.height = 4)
N <- current_data %>%
group_by(subject) %>%
summarize(n()) %>%
nrow()

frustrate_means <- current_data %>%
group_by(subject, difficulty, curve) %>%
summarize(frustrate = mean(frustrate, na.rm = TRUE)) %>%
group_by(difficulty, curve) %>%
summarize(frustrate_mean = mean(frustrate, na.rm = TRUE), frustrate_se = sd(fr
ustrate, na.rm = TRUE) / sqrt(N))

frustrate_means %>%
ggplot(aes(x = difficulty, y = frustrate_mean, group = curve)) +
geom_bar(stat = 'identity', aes(fill = curve), position = 'dodge') +
geom_errorbar(stat = 'identity', aes(ymin = frustrate_mean - frustrate_se, yma
x = frustrate_mean + frustrate_se),
              position = position_dodge(width = .9), width = .5) + ggtitle('Lik
elihood of Being Frustrated') +
xlab('Difficulty of Exam') + ylab('Likelihood of Being Frustrated') + ylim(0,6
)
```



It looks like, overall, people seem to expect they'd be pretty frustrated. It looks like we're seeing two main effects and no interaction. Let's check that intuition with an ANOVA.

ANOVA

In [8]:
```
frustrate_model <- ezANOVA(data = current_data[complete.cases(current_data$frustrate)],
                                                                    wid = subject, within = .(difficulty, curve),
                                                                    dv = frustrate,
 detailed = TRUE)
cbind(frustrate_model$ANOVA, n2p = frustrate_model$ANOVA$SSn / (frustrate_model$ANOVA$SSn + frustrate_model$ANOVA$SSd))
```

```
Warning message:
"Converting "subject" to factor for ANOVA."
```

Effect	DFn	DFd	SSn	SSd	F	p	p<.05	ges
(Intercept)	1	58	5164.4745763	427.52542	700.635585	4.502733e-34	*	0.8986
difficulty	1	58	51.2711864	75.72881	39.268129	4.937014e-08	*	0.0808
curve	1	58	63.0677966	48.93220	74.755109	5.111333e-12	*	0.0976
difficulty:curve	1	58	0.6101695	30.38983	1.164529	2.849957e-01		0.0010

Yup, just two main effects.

Means

In [9]:
```
frustrate_means
```

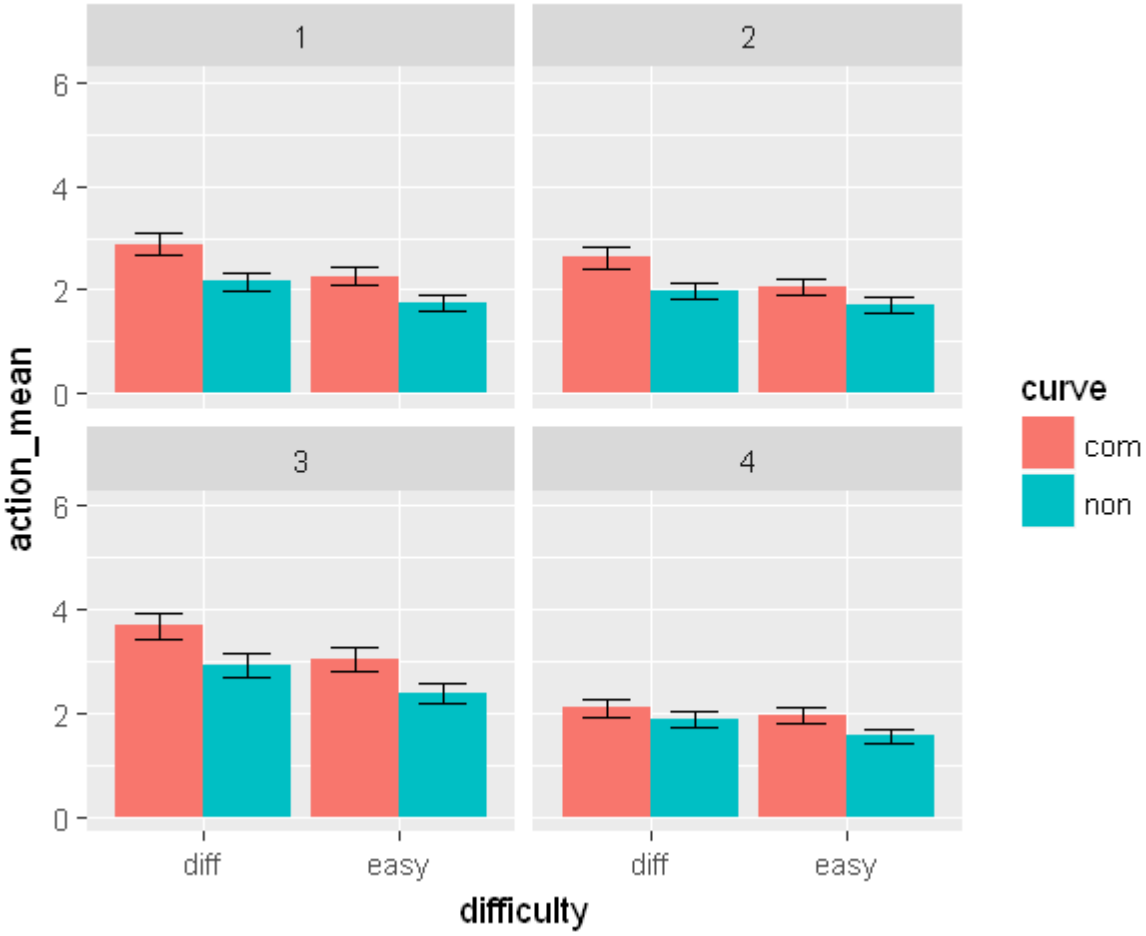
difficulty	curve	frustrate_mean	frustrate_se
diff	com	5.610169	0.2022910
diff	non	4.677966	0.2221523
easy	com	4.779661	0.1858020
easy	non	3.644068	0.2131870

If you guys want access to subject-level means for whatever reason, I'll make them all available in SPSS-friendly format near the end of the document.

Action Data

Plot

In [13]:
```
current_data %>%
group_by(difficulty, curve, question_number) %>%
summarize(action_mean = mean(action, na.rm = TRUE), action_sd = sd(action, na.
rm = TRUE) / sqrt(N))  %>%
ggplot(aes(x = difficulty, y = action_mean, group = curve)) +
geom_bar(stat = 'identity', aes(fill = curve), position = 'dodge') +
geom_errorbar(stat = 'identity', aes(ymin = action_mean - action_sd, ymax = ac
tion_mean + action_sd),
              position = position_dodge(width = .9), width = .5) + facet_wrap(~q
uestion_number) + ylim(0,6)
```



Just for reference:
Question 1:
How likely would you do something in the scenario above?

Question 2:
How likely would you report to the professor that this student cheated after class?

Question 3:
How likely would you anonymously report that this student cheated after class?

Question 4:
How likely would you confront the cheating student at the end of the exam to suggest they confess?

ANOVAs

Fasten your seatbelts, because we're about to run (at least) five ANOVAs.

Running an overall ANOVA to get the main effects

In [11]:
```
head(current_data)
```

subject	difficulty	curve	question_number	action	frustrate
1	diff	com	1	2	3
1	diff	com	2	3	NA
1	diff	com	3	3	NA
1	diff	com	4	3	NA
1	diff	non	1	3	2
1	diff	non	2	3	NA

In [19]:
```
m1 <- ezANOVA(data = current_data, wid = subject, within = .(difficulty, curve
), dv = action, detailed = TRUE)
cbind(m1$ANOVA, n2p = m1$ANOVA$SSn / (m1$ANOVA$SSn + m1$ANOVA$SSd))
```

```
Warning message:
"Converting "subject" 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	ges
(Intercept)	1	58	1258.5805085	288.57574	252.9584403	8.176733e-23	*	0.784
difficulty	1	58	11.4576271	21.26112	31.2562218	6.376001e-07	*	0.032
curve	1	58	16.5519068	25.72934	37.3118966	9.026656e-08	*	0.045
difficulty:curve	1	58	0.1790254	10.91472	0.9513272	3.334323e-01		0.000

So here you can see that, collapsing across question type, we get two main effects and no interaction.

Now we'll see whether or not the interaction is significant when broken down by question.

Running one ANOVA for each question

In [21]:
```
current_data %>%
group_by(question_number) %>%
do(data.frame(ezANOVA(data = ., wid = subject, within = .(difficulty, curve),
 dv = action, detailed = TRUE))) %>%
filter(ANOVA.Effect == 'difficulty:curve') %>%
mutate(n2p = ANOVA.SSn / (ANOVA.SSn + ANOVA.SSd))
```

```
Warning message:
"Converting "subject" to factor for ANOVA."Warning message:
"Converting "subject" to factor for ANOVA."Warning message:
"Converting "subject" to factor for ANOVA."Warning message:
"Converting "subject" to factor for ANOVA."
```

question_number	ANOVA.Effect	ANOVA.DFn	ANOVA.DFd	ANOVA.SSn	ANOVA.SSd	AN
1	difficulty:curve	1	58	0.7161017	33.03390	1.2
2	difficulty:curve	1	58	1.3728814	17.62712	4.5
3	difficulty:curve	1	58	0.1059322	21.14407	0.2
4	difficulty:curve	1	58	0.4237288	20.07627	1.2

Here are *only* the interaction terms from the four ANOVAs looking at the four question types. As you can see, the interaction is only significant for the second question.

Means

I'm going to report:

- The means for the two overall main effects
- The means for the two-way interaction for question 2
- The marginal means of action by question type

Means for two overall main effects

In [23]:
```
current_data %>%
group_by(subject, difficulty, curve) %>%
summarize(action = mean(action)) %>%
group_by(difficulty) %>%
summarize('Action (Mean)' = mean(action), 'Action (SE)' = sd(action ) / sqrt(N
))

current_data %>%
group_by(subject, difficulty, curve) %>%
summarize(action = mean(action)) %>%
group_by(curve) %>%
summarize('Action (Mean)' = mean(action), 'Action (SE)' = sd(action ) / sqrt(N
))
```

difficulty	Action (Mean)	Action (SE)
diff	2.529661	0.1738026
easy	2.088983	0.1496970

curve	Action (Mean)	Action (SE)
com	2.574153	0.1730790
non	2.044492	0.1480615

Means for two-way interaction for question 2

In [24]:
```
current_data %>%
filter(question_number == 2) %>%
group_by(subject, difficulty, curve) %>%
summarize(action = mean(action)) %>%
group_by(difficulty, curve) %>%
summarize('Action (Mean)' = mean(action), 'Action (SE)' = sd(action ) / sqrt(N
))
```

difficulty	curve	Action (Mean)	Action (SE)
diff	com	2.627119	0.2106158
diff	non	1.983051	0.1575891
easy	com	2.050847	0.1647195
easy	non	1.711864	0.1410714

Marginal means for action by question type

In [28]:
```
current_data %>%
group_by(subject, difficulty, curve, question_number) %>%
summarize(action = mean(action)) %>%
group_by(question_number) %>%
summarize('Action (Mean)' = mean(action), 'Action (SE)' = sd(action) / sqrt(N
))
```

question_number	Action (Mean)	Action (SE)
1	2.266949	0.1849323
2	2.093220	0.1748626
3	3.004237	0.2406560
4	1.872881	0.1598292

Overall, people are most likely to act for question three, and least likely for question 4.

Data Access

I'm going to save data to the "data" folder in your guys' file in the OSF so that you can access it if you want; more specifically: `./Integrity Group/data`

`subject_action.xlsx` -- The means for the full design (including question type) for action broken down by subject
`subject_frustrate.xlsx` -- The means for the full design for frustration broken down by subject

In [39]:
```
library(reshape2)
current_data %>%
group_by(subject, difficulty, curve, question_number) %>%
summarize(action = mean(action)) %>%
dcast(subject ~ difficulty + curve + question_number) %>%
mutate(subject = as.numeric(subject)) %>%
arrange(subject) %>%
write.xlsx('../data/subject_action.xlsx', row.names = FALSE)

current_data %>%
group_by(subject, difficulty, curve)  %>%
summarize(frustrate = mean(frustrate, na.rm = TRUE)) %>%
dcast(subject ~ difficulty + curve) %>%
mutate(subject = as.numeric(subject)) %>%
arrange(subject) %>%
write.xlsx('../data/subject_frustrate.xlsx', row.names = FALSE)
```

```
Using action as value column: use value.var to override.
Using frustrate as value column: use value.var to override.
```