Music Insights

Olivia Lund

First, we need tidyverse to handle data importing and wrangling…

```
#install.packages("tidyverse")

survey <- read.csv("https://raw.githubusercontent.com/introdsci/MusicSurvey/master/mu
sic-survey.csv")
preferences <- read.csv("https://raw.githubusercontent.com/introdsci/MusicSurvey/mast
er/preferences-survey.csv")
```

The data in the music survey table has inconsistent and verbose naming that isn’t conducive to good datakeeping. We can change the column names to better represent Cleaning data frame column names:

```
colnames(survey)[colnames(survey)=="Timestamp"]<-"time_submitted"
colnames(survey)[colnames(survey)=="First..we.are.going.to.create.a.pseudonym.for.yo
u.to.keep.this.survey.anonymous..more.or.less...Which.pseudonym.generator.would.you.p
refer."]<-"pseudonym_generator"
colnames(survey)[colnames(survey)=="What.is.your.pseudonym."]<-"pseudonym"
colnames(survey)[colnames(survey)=="Sex"]<-"sex"
colnames(survey)[colnames(survey)=="Major"]<-"academic_major"
colnames(survey)[colnames(survey)=="Academic.Year"]<-"academic_level"
colnames(survey)[colnames(survey)=="Year.you.were.born..YYYY."]<-"year_born"
colnames(survey)[colnames(survey)=="Which.musical.instruments.talents.do.you.play...S
elect.all.that.apply."]<-"instrument_list"
colnames(survey)[colnames(survey)=="Artist"]<-"favorite_song_artist"
colnames(survey)[colnames(survey)=="Song"]<-"favorite_song"
colnames(survey)[colnames(survey)=="Link.to.song..on.Youtube.or.Vimeo."]<-"favorite_s
ong_link"

colnames(survey)
```

```
##  [1] "time_submitted"       "pseudonym_generator"  "pseudonym"
##  [4] "sex"                  "academic_major"       "academic_level"
##  [7] "year_born"            "instrument_list"      "favorite_song_artist"
## [10] "favorite_song"        "favorite_song_link"
```

```
#colnames(survey)[colnames(survey)=="timestamp"]<-"time_submitted" #sidenote: this do
esn't work. Why?
```

This dataset completely encapsulates the data, but it describes several types of things, and has columns that describe multiple pieces of information, violating the principles of tidy data. In order to resolve this, the table will need to be neatly divided.

```
library(tidyverse)
```

```
## ── Attaching packages ──────────────────────────────────── tidyverse 1.2.1 ──
## ✔ ggplot2 3.2.1     ✔ purrr   0.3.2
## ✔ tibble  2.1.3     ✔ dplyr   0.8.3
## ✔ tidyr   1.0.0     ✔ stringr 1.4.0
## ✔ readr   1.3.1     ✔ forcats 0.4.0
```

```
## ── Conflicts ─────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library("dplyr")
```

First, let’s create a person table with each of the responses from the survey. This table should include pseudonym_generator_name, pseudonym, sex, academic_major, academic_level, and year_born:

```
person<-tibble(pseudonym_generator = survey$pseudonym_generator,  pseudonym = survey
$pseudonym, sex = survey$sex, academic_major = survey$academic_major, academic_level
= survey$academic_level, year_born = survey$year_born)
colnames(person)
```

```
## [1] "pseudonym_generator" "pseudonym"           "sex"
## [4] "academic_major"      "academic_level"      "year_born"
```

Next, in order to store the data that we just took out, let’s create a survey table that contains pseudonym, and the name, artist, and url of the person’s favorite song.

```
favorite_song<-tibble(pseudonym = survey$pseudonym, favorite_song = survey$favorite_s
ong, favorite_song_artist = survey$favorite_song_artist, favorite_song_link = survey
$favorite_song_link)
colnames(favorite_song)
```

```
## [1] "pseudonym"            "favorite_song"        "favorite_song_artist"
## [4] "favorite_song_link"
```

Our dataset contains categorical data that we want to remake into discrete data, specifically a factor. To do this, we need to access and modify each of the possible options in that factor:

```
person$academic_level<-as_factor(person$academic_level)
levels(person$academic_level)
```

```
## [1] "Junior"    "Senior"    "Sophomore"
```

```
person$academic_major<-as_factor(person$academic_major)
levels(person$academic_major)[levels(person$academic_major) == "Computer  Engineering
"] <- "Computer Engineering"
levels(person$academic_major)[levels(person$academic_major) == "Computer information
 systems"] <- "Computer Information Systems"
levels(person$academic_major)
```

```
## [1] "Computer Engineering "        "Computer Information Systems"
## [3] "Computer Science"             "Math"
```

We want to take the table of song preferences and use the tidyverse gather function to produce a much tidier table that has the pseudonym artist_song, and rating for every single song:

```
colnames(preferences)[colnames(preferences)=="What.was.your.pseudonym."]<-"pseudonym"
preferences$Timestamp <- NULL
ratings <- gather(preferences, song_name, rating, "X40.crew.Not.Enough":"Wheezer.Buddy.Holly")
```

Next, we want to make a table of the musical talents, lsiting each as a separate row for each person, and combining all the different phrasings combined by talent keyword.

```
talents <-tibble(pseudonym = survey$pseudonym, talent = survey$instrument)
talents <-talents %>% separate_rows(talent, pseudonym, sep = ", ", convert = TRUE)
#subset(ChickWeight, Diet==4 && Time == 21)
talents$talent <- ifelse(grepl("piano|Piano", talents$talent), "Piano", talents$talent)
talents$talent <- ifelse(grepl("Ukelele", talents$talent), "Ukelele", talents$talent)
```