Week 3 solutions

Before we start, let’s load some packages and some data.

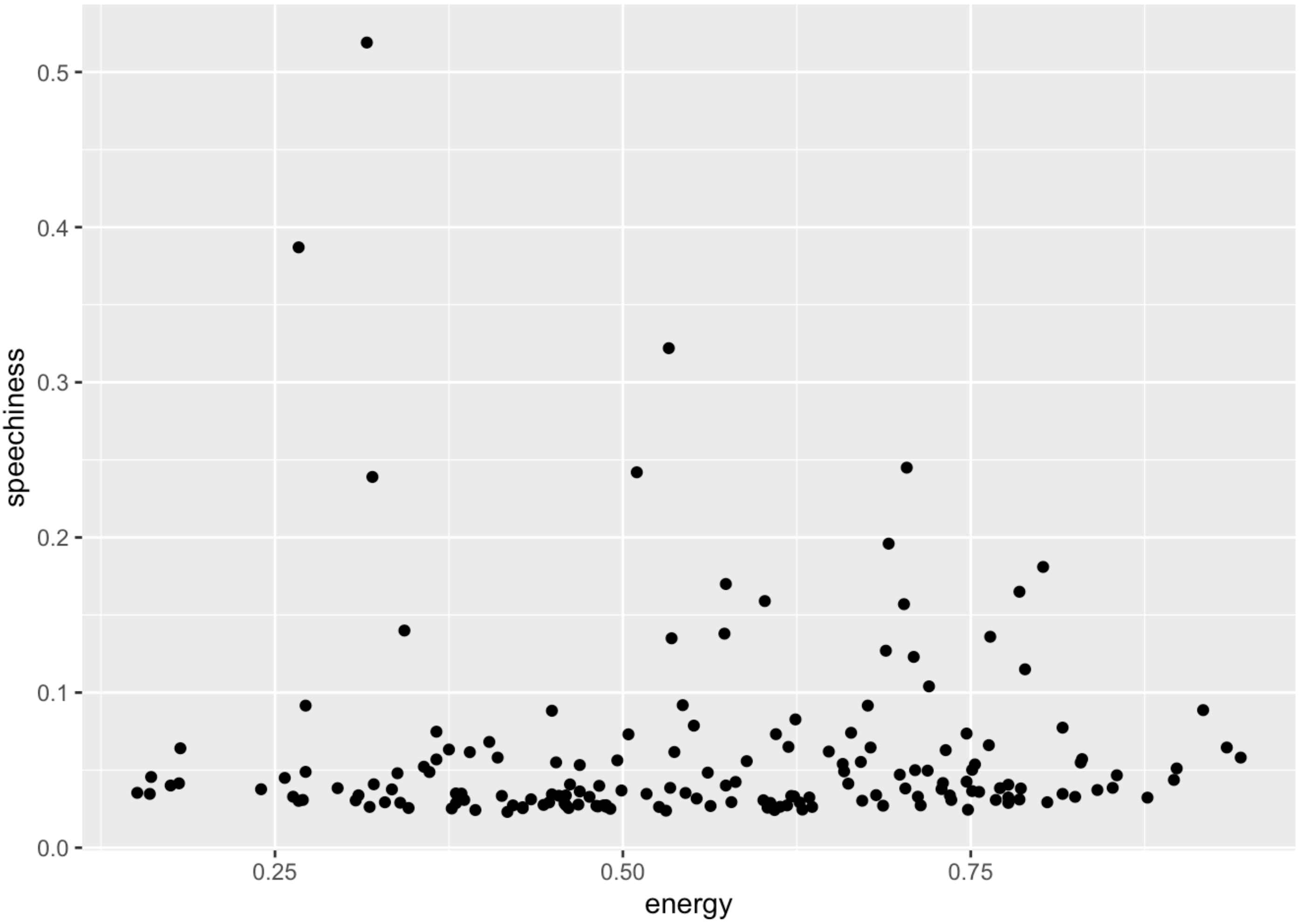
```
library(tidyverse)
library(lubridate)
```

```
taylor <- read_csv("https://bit.ly/swifty-data")
beyonce <- read_csv("https://bit.ly/beyonce-data-csv")
monkeys <- read_csv("https://bit.ly/arctic-monkeys-data")
```

Q1: What’s the relationship between energy and speechiness in Taylor Swift’s albums?

```
ggplot(data = taylor) +
  aes(x = energy, y = speechiness) +
  geom_point()
```

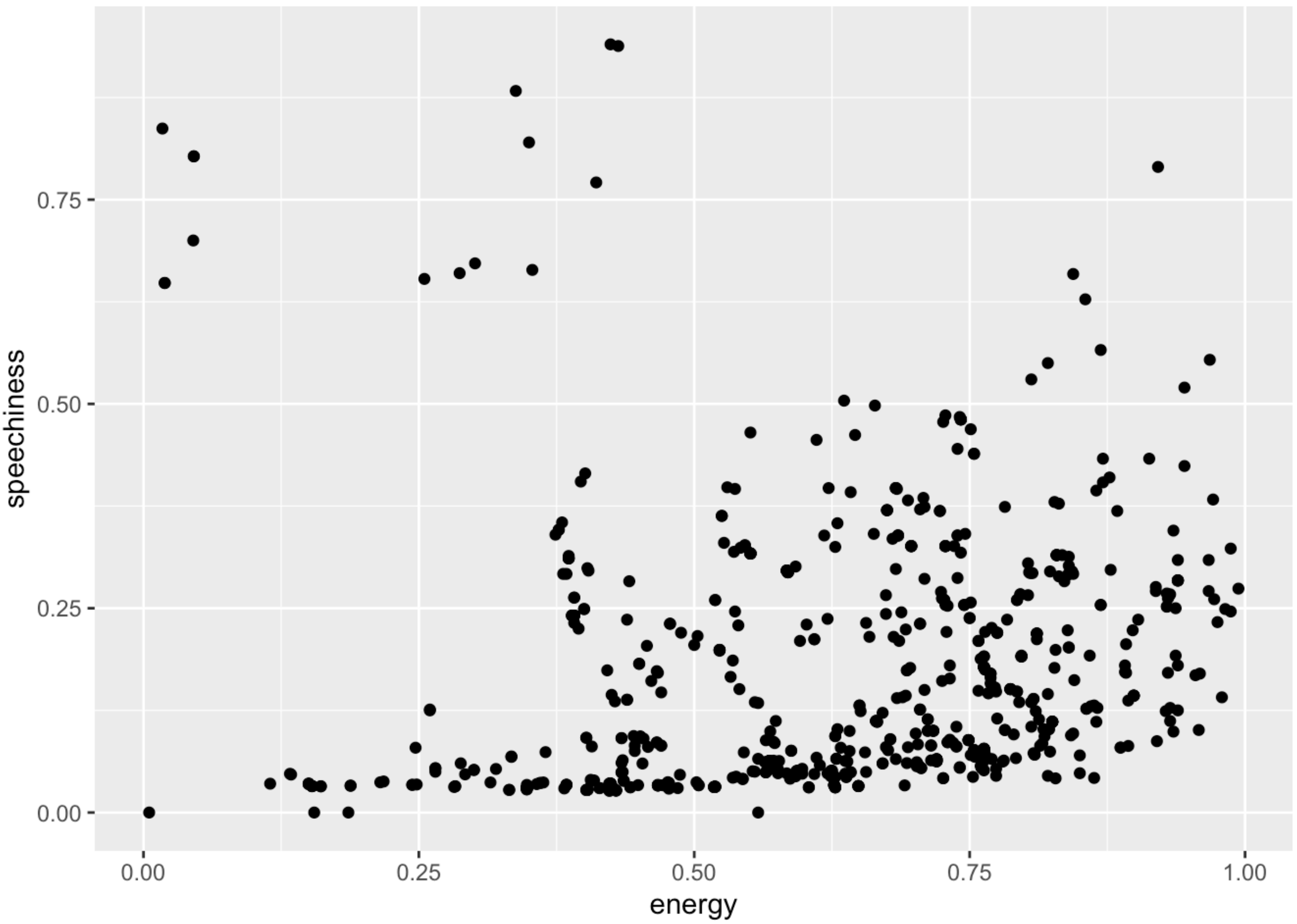
```
## Warning: Removed 12 rows containing missing values or values outside the scale
range
## (`geom_point()`).
```



We can see that very few of Taylor Swift's songs have high speechiness, but there's one that's very high, up at the top left.

Q2: …and Beyonce's?

```
ggplot(data = beyonce) +
  aes(x = energy, y = speechiness) +
  geom_point()
```



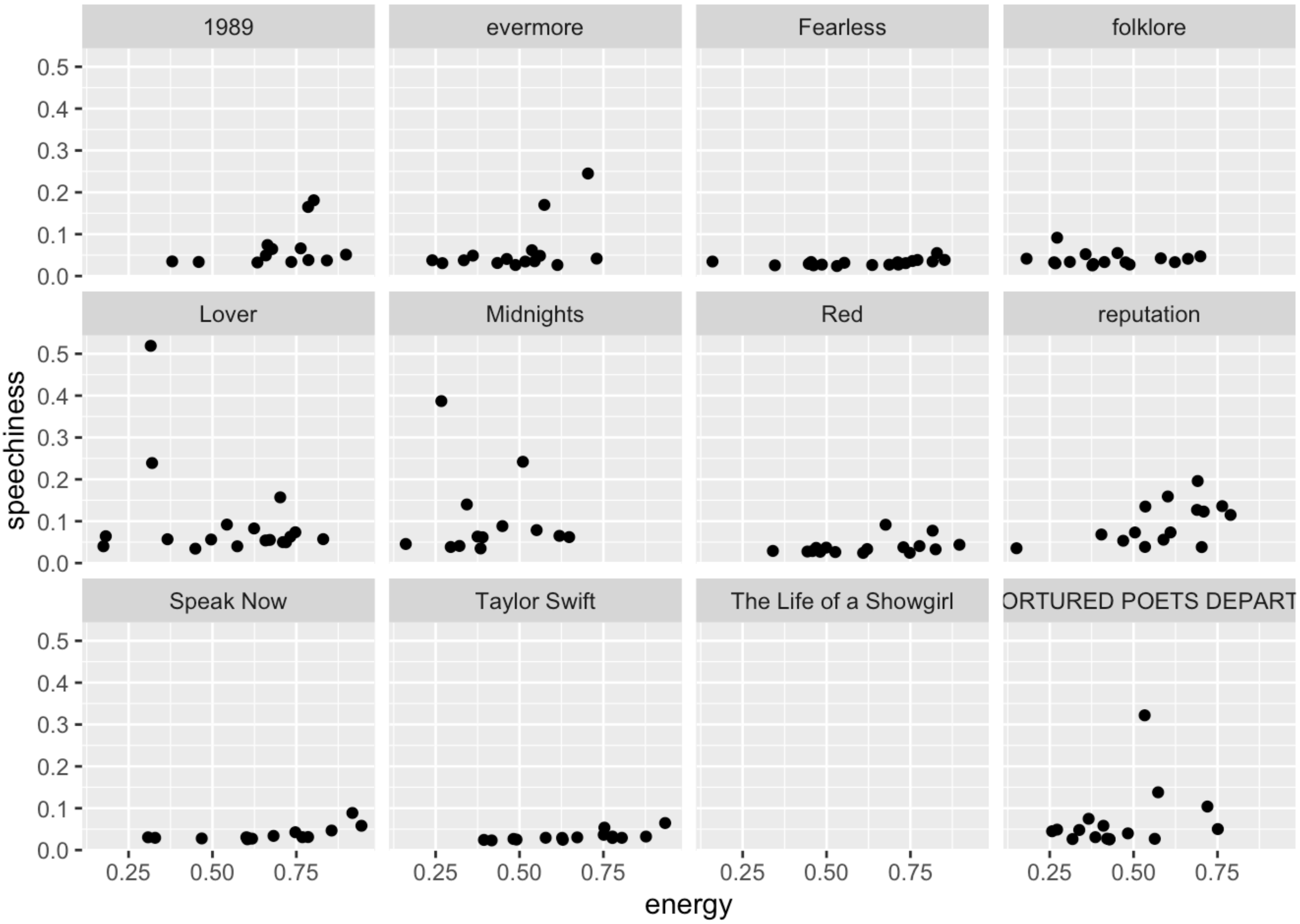
This is the figure that you generate when you use all of Beyonce's songs. You can see there's an "island" or "cluster" of sorts up at the top left with tracks with high speechiness and low energy.

Q3: How do these relationships vary by album?

For simplicity, let's look at Taylor Swift again.

```
ggplot(data = taylor) +
  aes(x = energy, y = speechiness) +
  geom_point() +
  facet_wrap(~ album)
```

```
## Warning: Removed 12 rows containing missing values or values outside the scale
range
## (`geom_point()`).
```

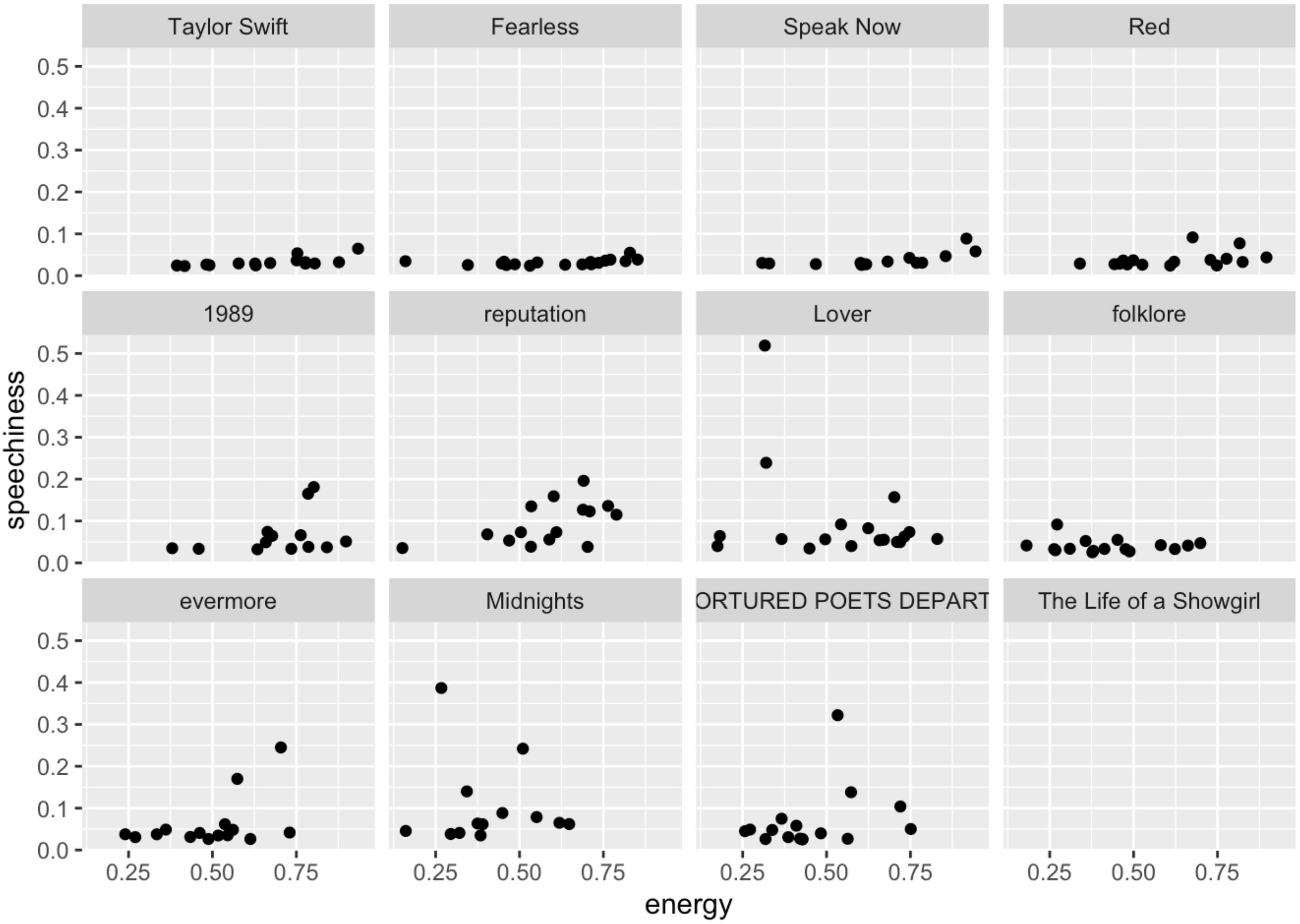


This shows us that the majority of Taylor Swift's albums have super-low speechiness. However, as before, it's hard to interpret this data give the albums are in alphabetical order. Let's reorganise the plot so we can see how this has changed over her career, by generating a **levels** object again.

```
taylor_levels <- c(
  "Taylor Swift",
  "Fearless",
  "Speak Now",
  "Red",
  "1989",
  "reputation",
  "Lover",
  "folklore",
  "evermore",
  "Midnights",
  "THE TORTURED POETS DEPARTMENT",
  "The Life of a Showgirl"
  )

ggplot(data = taylor) +
  aes(x = energy, y = speechiness) +
  geom_point() +
  facet_wrap(~ factor(album,
                      levels = taylor_levels))
```

```
## Warning: Removed 12 rows containing missing values or values outside the scale
range
## (`geom_point()`).
```



This shows that her earlier albums had super-low speechiness, while her more recent albums have been more of a mix of high and low speechiness, with Lover featuring the track we saw earlier with super-high speechiness.

Q4: What about for the Arctic Monkeys?

In this case, I asked you to put the albums in chronological order. Let’s have a look at the data:

```
head(monkeys)
```

```
## # A tibble: 6 × 27
##   artist   disc_number duration_ms explicit href  id    name  track_number type
##   <chr>          <dbl>       <dbl> <lgl>    <chr> <chr> <chr>        <dbl> <chr
>
## 1 Arctic …           1      265798 FALSE    http… 1zx6… Ther…            1 trac
k
## 2 Arctic …           1      191173 FALSE    http… 1UwU… I Ai…            2 trac
k
## 3 Arctic …           1      239151 FALSE    http… 5hlj… Scul…            3 trac
k
## 4 Arctic …           1      197576 FALSE    http… 2HRe… Jet …            4 trac
k
## 5 Arctic …           1      290584 FALSE    http… 42Gu… Body…            5 trac
k
## 6 Arctic …           1      198554 FALSE    http… 0C6d… The …            6 trac
k
## # ℹ 18 more variables: uri <chr>, is_local <lgl>, external_urls.spotify <chr>,
## #   album <chr>, release_date <date>, recco_id <chr>, acousticness <dbl>,
## #   danceability <dbl>, energy <dbl>, instrumentalness <dbl>, key <chr>,
## #   liveness <dbl>, loudness <dbl>, mode <chr>, speechiness <dbl>, tempo <dbl>,
## #   valence <dbl>, popularity <dbl>
```

We can specify our levels in the same way we did with the Taylor Swift dataset. However, we need to pay careful attention to the case sensitivity and the punctuation, because it we don't write the album titles exactly the same way as they are written in the data, we won't be able to reorder them correctly.

It can be helpful to print out a list of all of the unique album names using the code below:

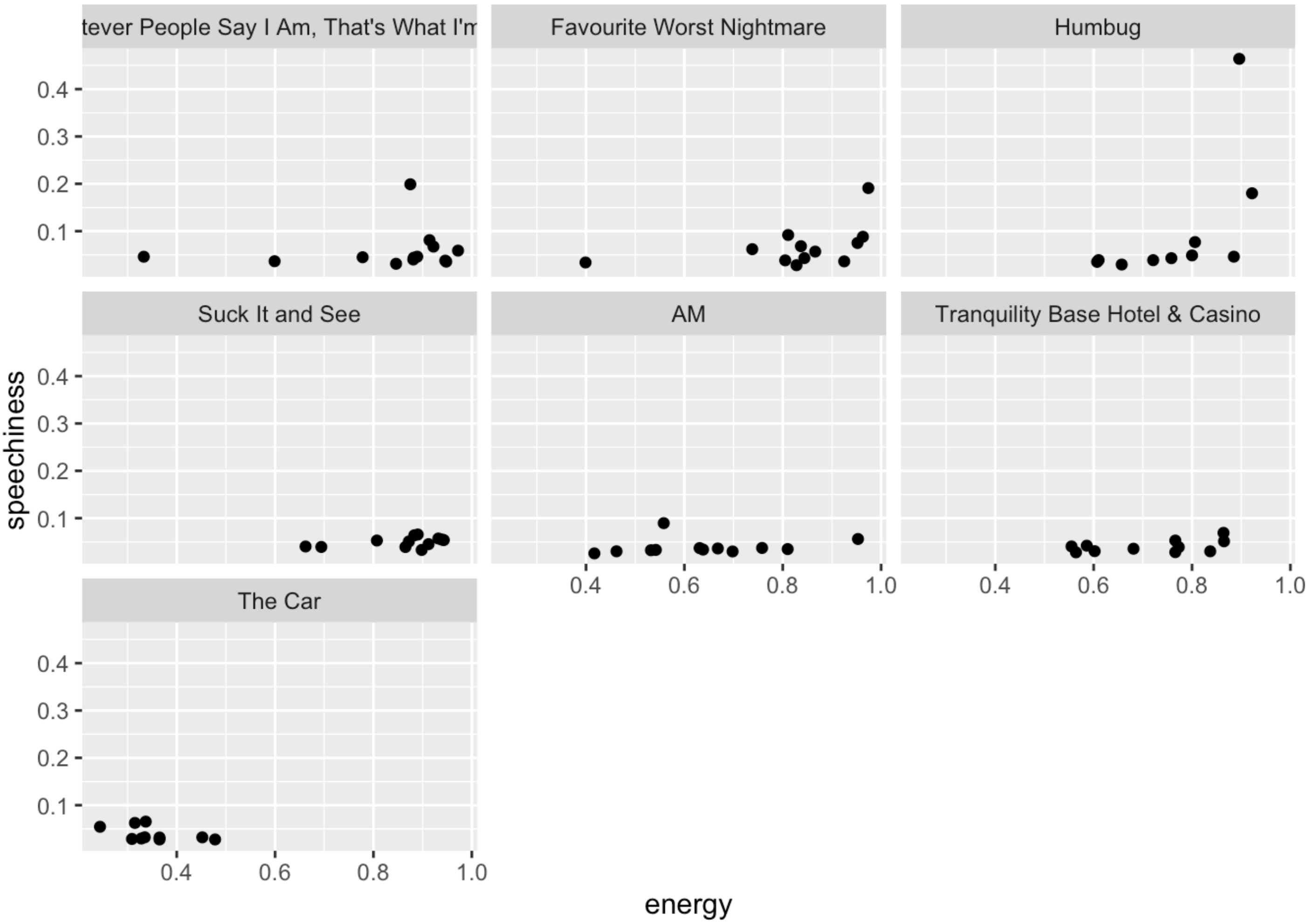
```
unique(monkeys$album)
```

```
## [1] "The Car"
## [2] "Tranquility Base Hotel & Casino"
## [3] "AM"
## [4] "Suck It and See"
## [5] "Humbug"
## [6] "Favourite Worst Nightmare"
## [7] "Whatever People Say I Am, That's What I'm Not"
```

In this case, our task is quite simple as the albums are in reverse chronological order already.

```
am_levels <-
  c("Whatever People Say I Am, That's What I'm Not",
    "Favourite Worst Nightmare",
    "Humbug",
    "Suck It and See",
    "AM",
    "Tranquility Base Hotel & Casino",
    "The Car")

ggplot(data = monkeys) +
  aes(x = energy, y = speechiness) +
  geom_point() +
  facet_wrap(~ factor(album,
                      levels = am_levels))
```

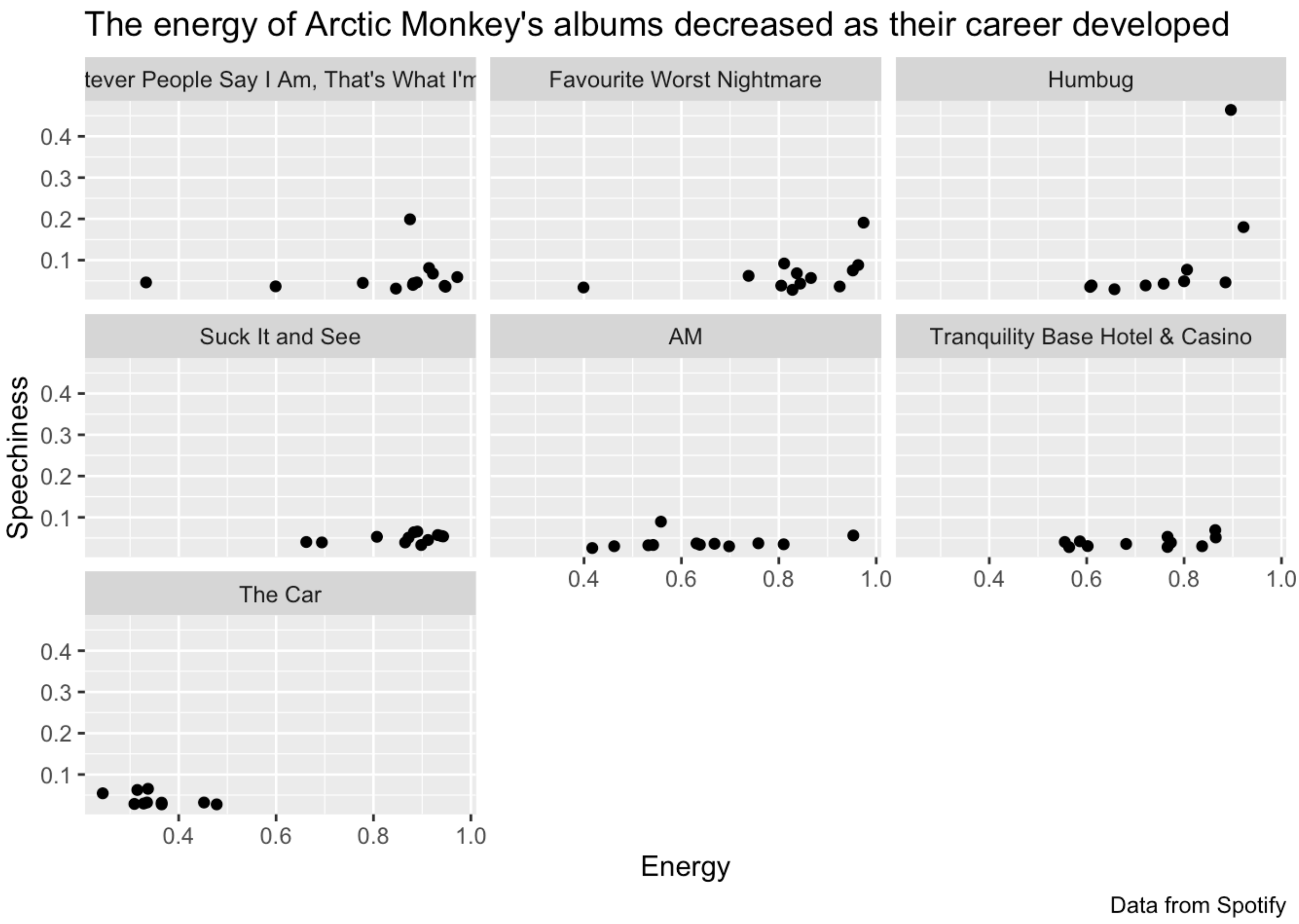


We can see at least one extreme outlier here – there is one track with extremely high speechiness on the Humbug Album (Pretty Visitors). While all of the albums have quite low speechiness scores, we can see that the energy of the average Arctic Monkeys album has changed quite a lot as their career has gone on! Their earlier albums were all quite high in energy, but AM marked a significant change and their most recent album, The Car, has incredibly low speechiness and energy.

Let’s clean up

We need to add labels! Let’s do that now.

```
ggplot(data = monkeys) +
  aes(x = energy, y = speechiness) +
  geom_point() +
  facet_wrap(~ factor(album,
                      levels = am_levels)) +
  labs(x = "Energy",
       y = "Speechiness",
       title = "The energy of Arctic Monkey's albums decreased as their career dev
eloped",
       caption = "Data from Spotify")
```



General feedback

- Remember to make your submissions ‘stand-alone’, with all of the code required to replicate them. For example, if you use a vector called ‘taylor_levels’, make sure you include the code used to create that vector in your submission.
- When using technical terms to interpret the plot, you might also want to think about how you could interpret the plot for a more general audience (e.g. how would you describe it in a blog post for Taylor Swift fans compared to to academics?)
- Remember to include your interpretation of the plots — it’s common to underestimate how much practice is needed picking out the interesting things from data visualisations.
- Remember that outliers and clusters (‘islands’) can be interesting things to point out when interpreting data visualisations, not least because they are the most obvious in data visualisations

and the least obvious when they are ‘hiding’ in the data itself.

- Not everyone had a go at practicing adding labels to their plots — I’d encourage you to have a go at this because it can help you learn how to write good active titles, etc.