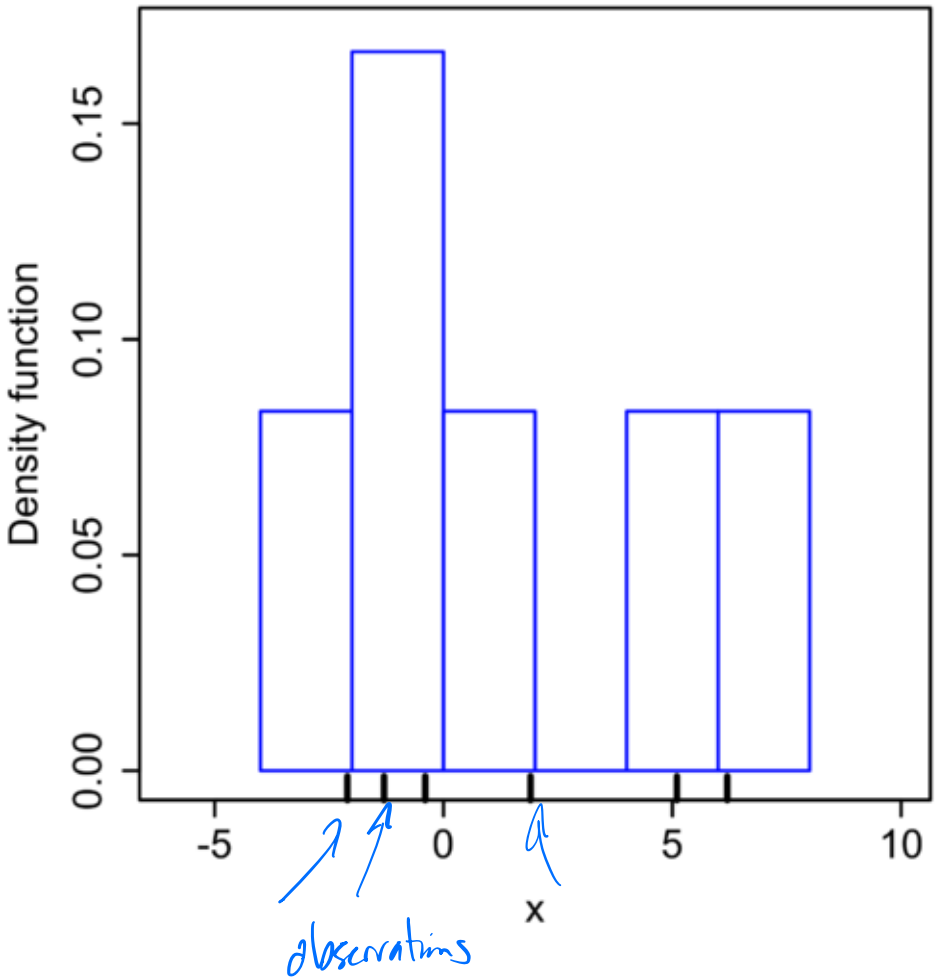
Kernel density estimation

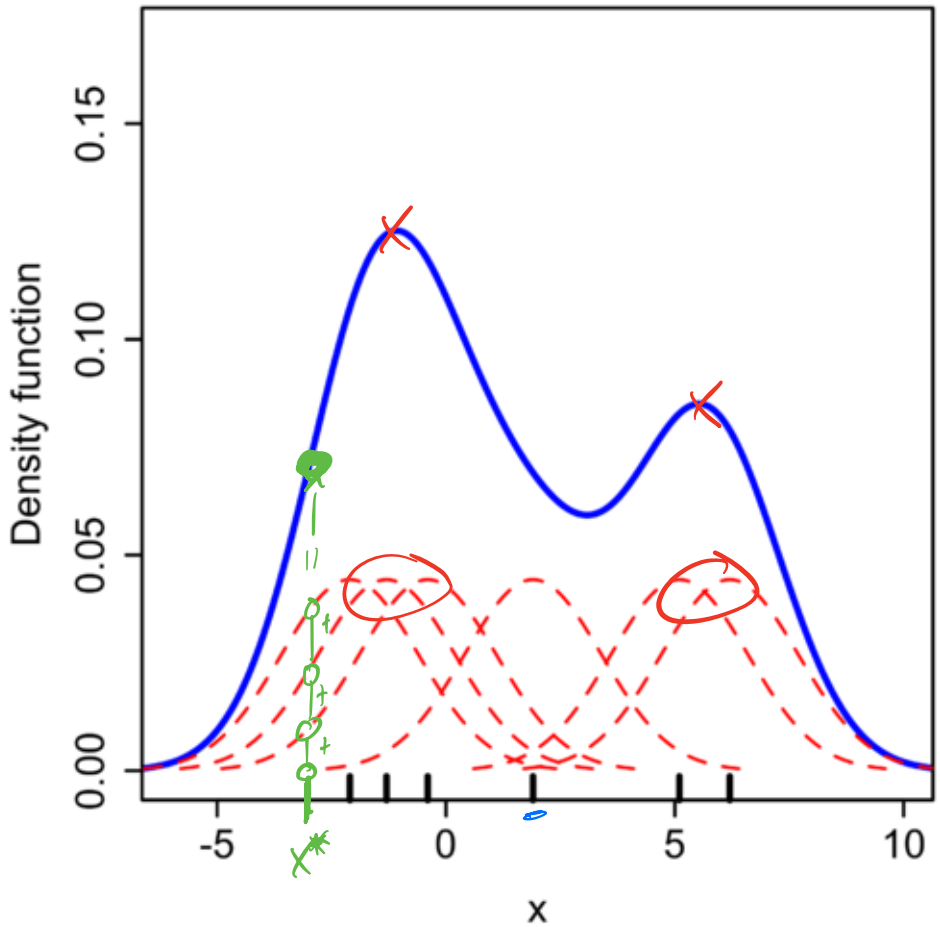
Goal: estimate PDF $f(x)$ for all possible values (assuming it is continuous & smooth)

$$\text{Kernel density estimate: } \hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h}K_h(x - x_i)$$

- $n =$ sample size, $x =$ new point to estimate $f(x)$ (does NOT have to be in dataset!)
- $h =$ **bandwidth**, analogous to histogram bin width, ensures $\hat{f}(x)$ integrates to 1
- $x_i = i$th observation in dataset
- $K_h(x - x_i)$ is the **Kernel** function, creates **weight** given distance of ith observation from new point
  - as $|x - x_i| \to \infty$ then $K_h(x - x_i) \to 0$, i.e. further apart ith row is from x, smaller the weight
  - as **bandwidth** $h \uparrow$ weights are more evenly spread out (as $h \downarrow$ more concentrated around x)
  - typically use **Gaussian** / Normal kernel: $\propto e^{-(x-x_i)^2/2h^2}$
  - $K_h(x - x_i)$ is large when x_i is close to x

Wikipedia example





We display kernel density estimates with `geom_density()`

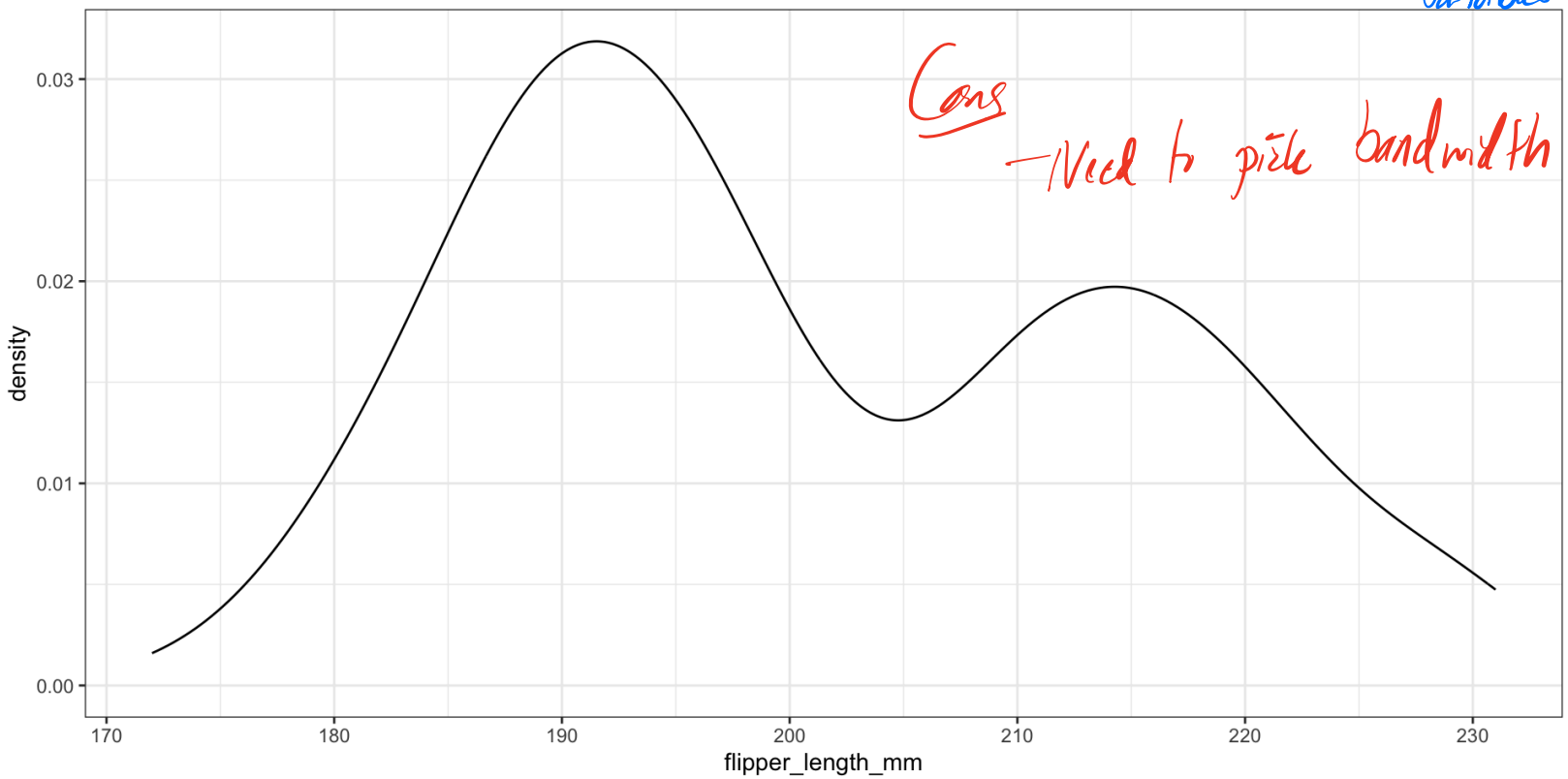
```
penguins |>
  ggplot(aes(x = flipper_length_mm)) +
  geom_density() +
  theme_bw()
```

Pros

+ Display full shape of distribution
+ Easily layer, add categorical variables w/ color

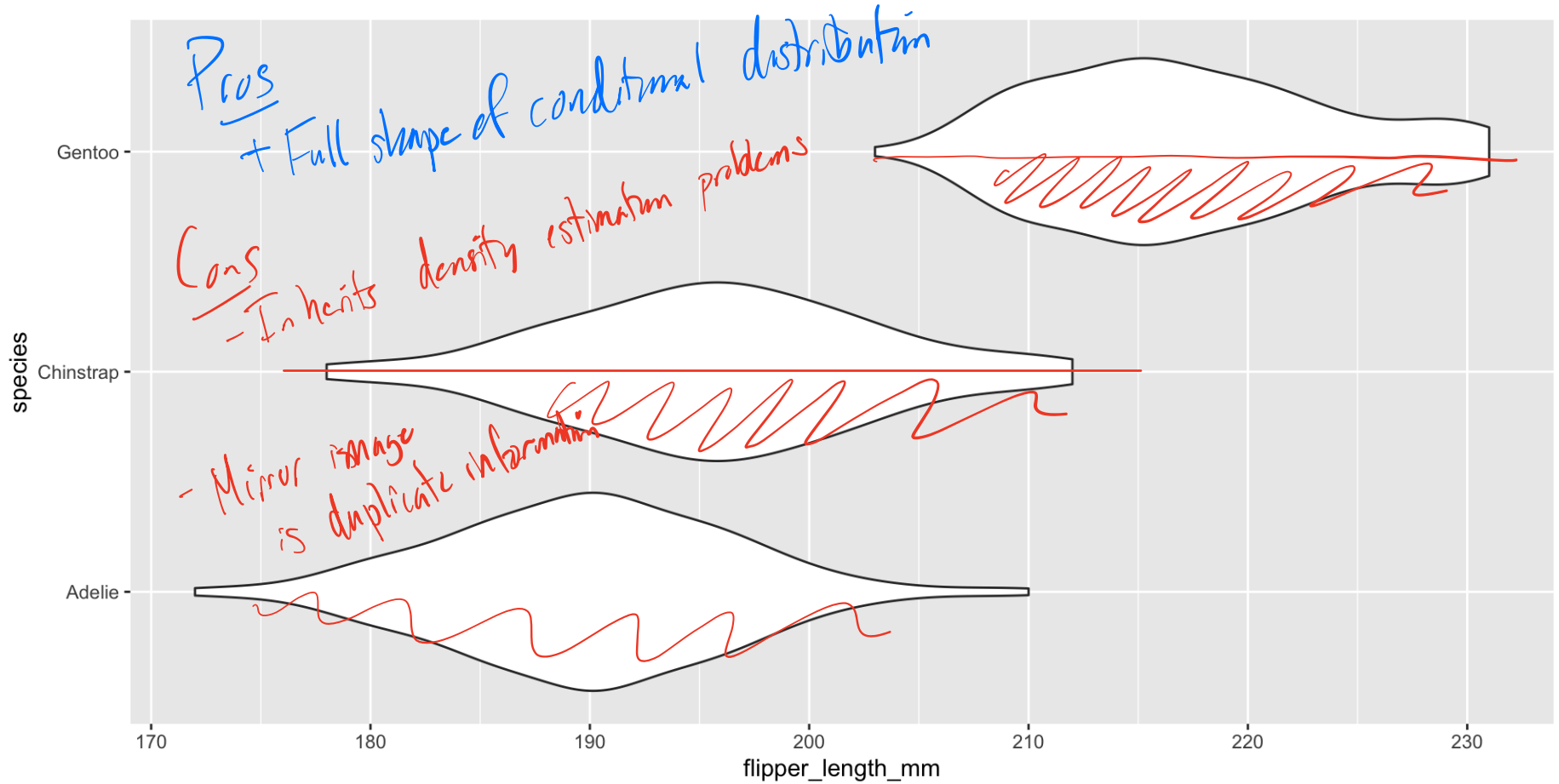
Cons

- Need to pick bandwidth & kernel



Visualizing conditional distributions: violin plots

```
penguins |>
  ggplot(aes(x = species, y = flipper_length_mm)) +
  geom_violin() +
  coord_flip()
```



Visualizing conditional distributions: ggbeeswarm

```
1 library(ggbeeswarm)
2 penguins |>
3   ggplot(aes(x = flipper_length_mm, y = species)) +
4   geom_beeswarm(cex = 1.5) +
5   theme_bw()
```