Week 2: Data Summary and Visualization

5. Putting it together

Stat 140 - 04

Mount Holyoke College

Dr. Shan Shan

Slides posted at *http://sshanshans.github.io/stat140*

Outline

1. Today: Comparing numerical variables across groups

2. Main ideas
    1. Visualization with ggplot
    2. Data wrangling

Outline

1. Today: Comparing numerical variables across groups

2. Main ideas
    1. Visualization with ggplot
    2. Data wrangling

Outline

1. Today: Comparing numerical variables across groups

2. Main ideas
    1. Visualization with ggplot
    2. Data wrangling

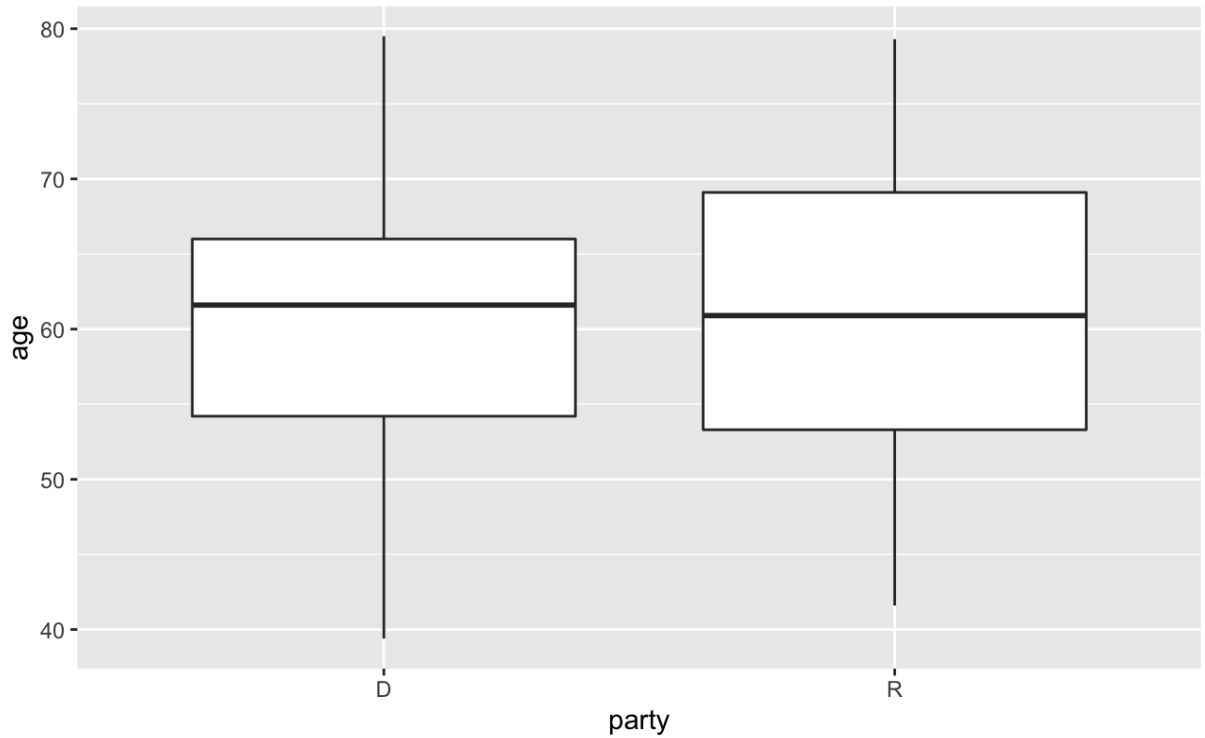
Some of the more interesting investigations can be considered by examining numerical data across groups.

The methods required here aren't really new: all that's required is to make a numerical plot for each group in the same graph.

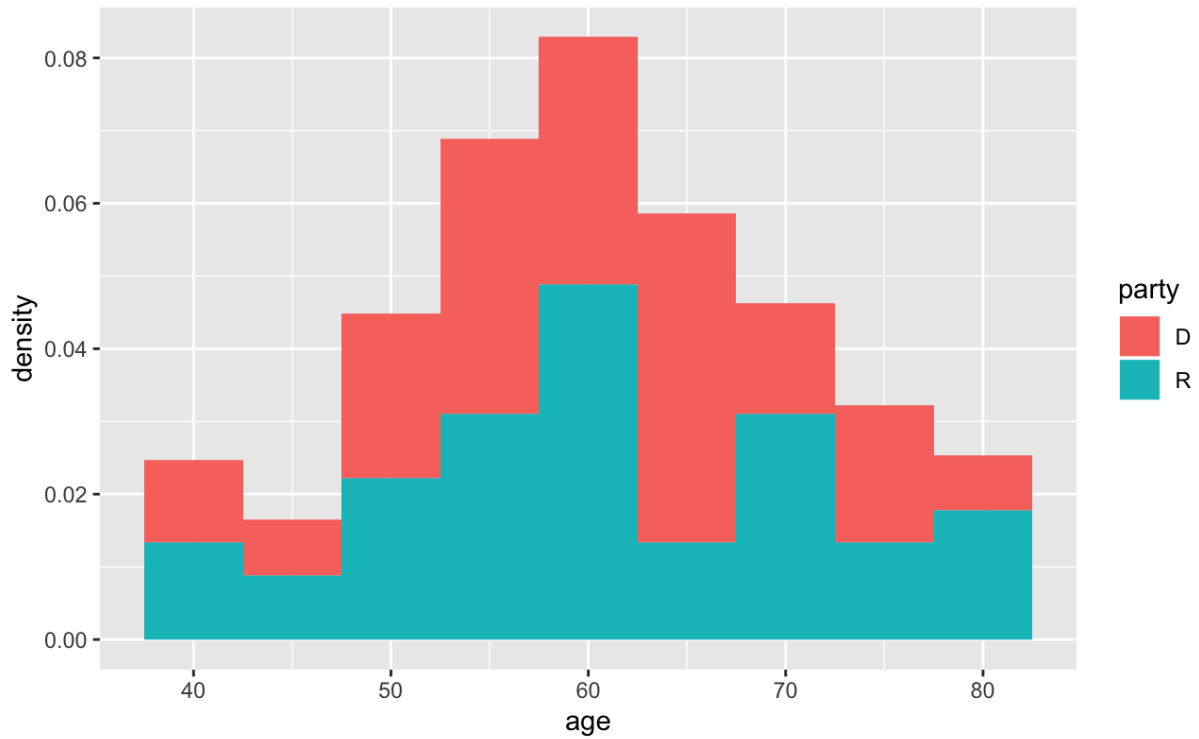
Here two convenient methods are introduced:

1. side-by-side box plot
2. colored histograms

Recall the data set called ‘senate 113‘ with information about the senators in the 113th US Senate. Below is a side by side box plot of the ‘age‘ variable across two parties: Democrats and Republicans.



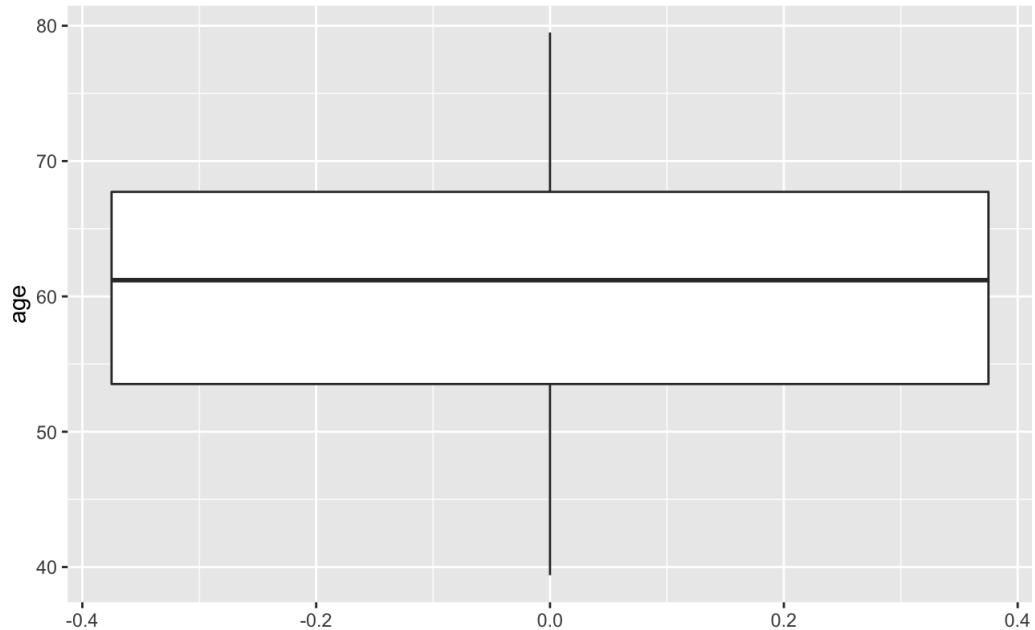
Recall the data set called ‘senate 113‘ with information about the senators in the 113th US Senate. Below is a colored histogram of the ‘age‘ variable across two parties: Democrats and Republicans.



Code:

```
ggplot(data = senate_113, mapping = aes(y = age)) +
    geom_boxplot()
```

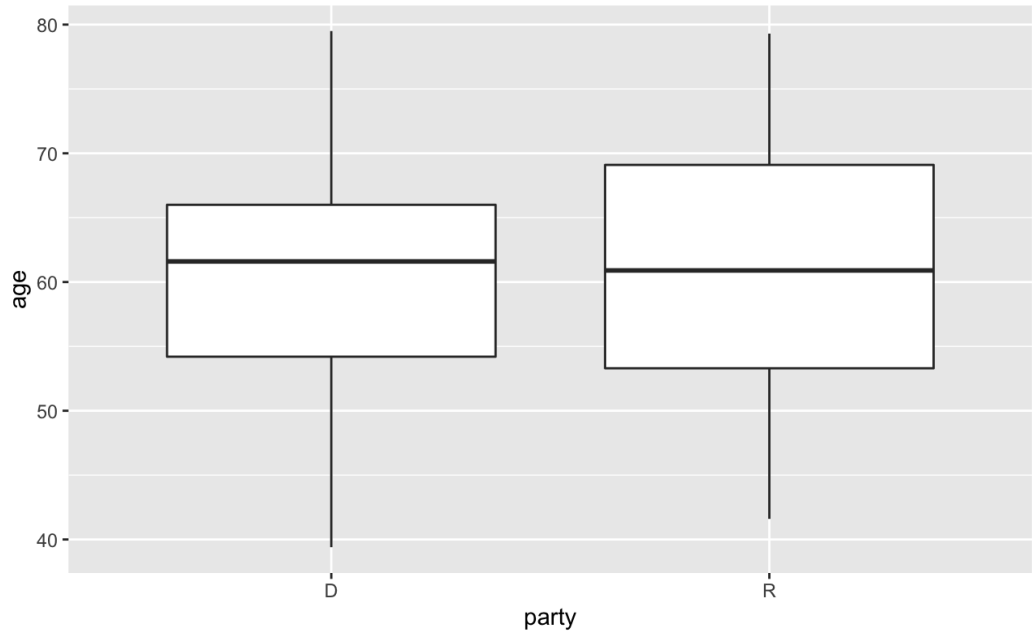
Plot:



Code:

```
ggplot(data = senate_113, mapping = aes(x = party, y = age))
+ geom_boxplot()
```

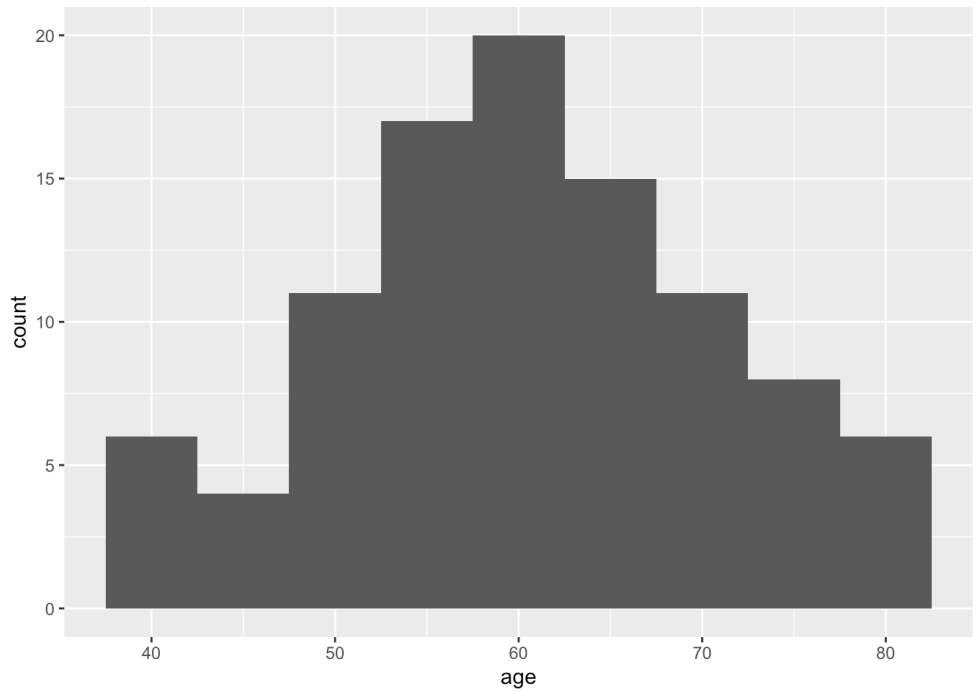
Plot:



Code:

```
ggplot(data = senate_113, mapping = aes(x = age)) +
    geom_histogram(binwidth = 5)
```

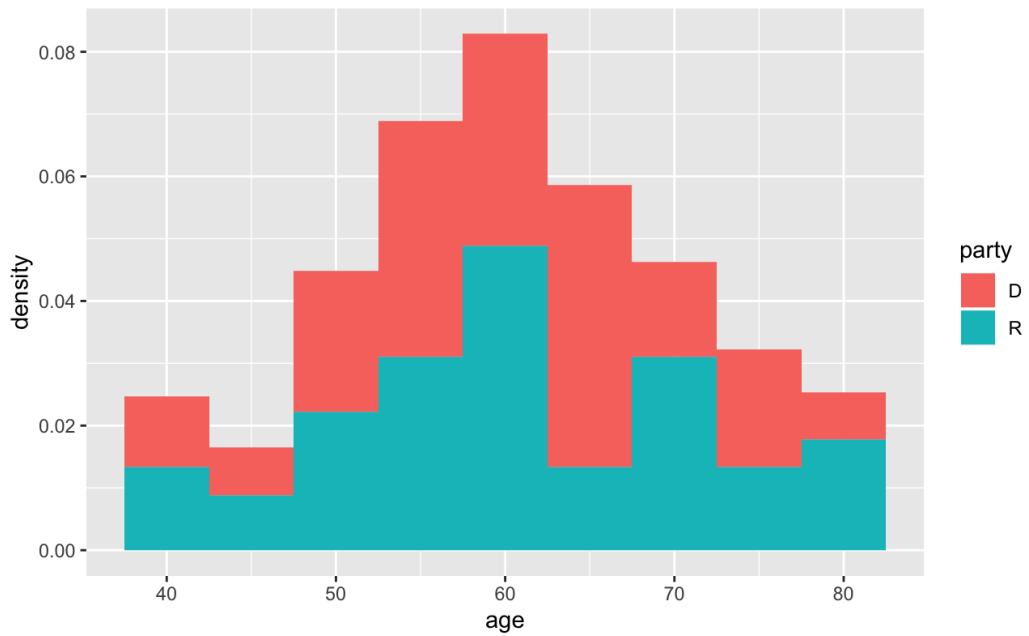
Plot:



Code:

```
ggplot(data = senate_113, mapping = aes(x = age, fill = party)) + geom_histogram(binwidth = 5)
```

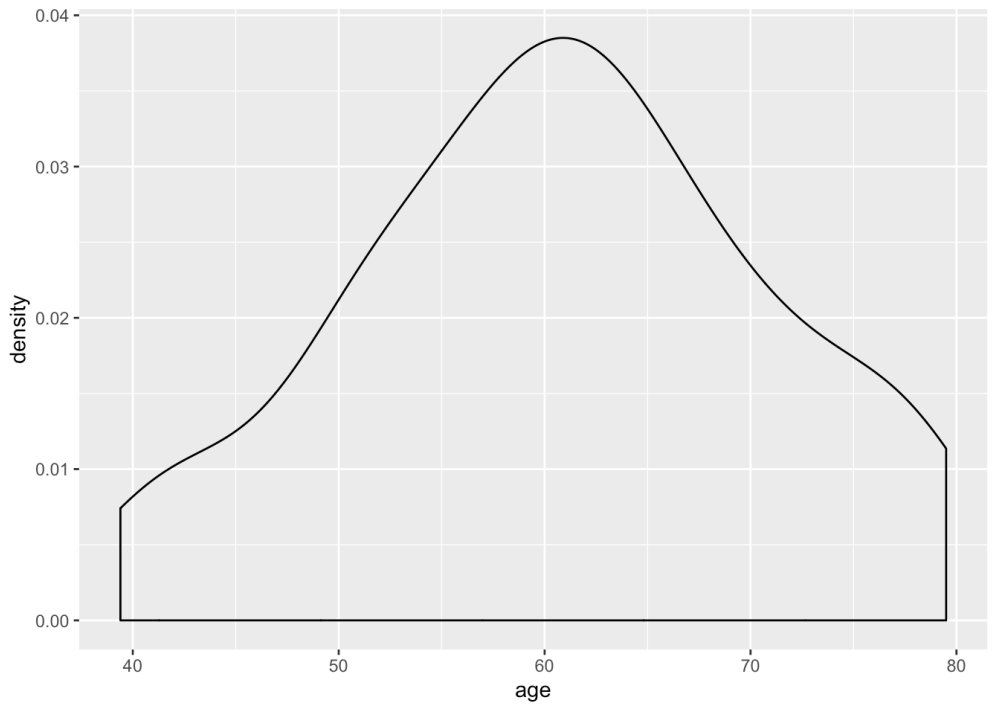
Plot:



Code:

```
ggplot(data = senate_113, mapping = aes(x = age)) +
    geom_density()
```

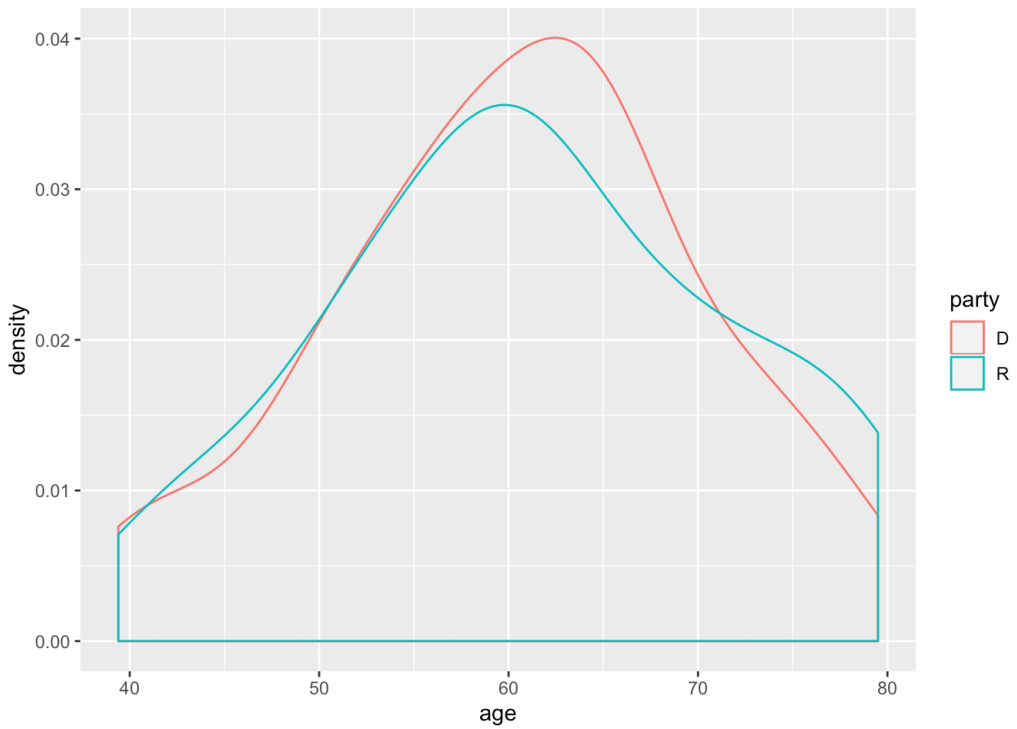
Plot:



Code:

```
ggplot(data = senate_113, mapping = aes(x = age, color = party)) + geom_density()
```

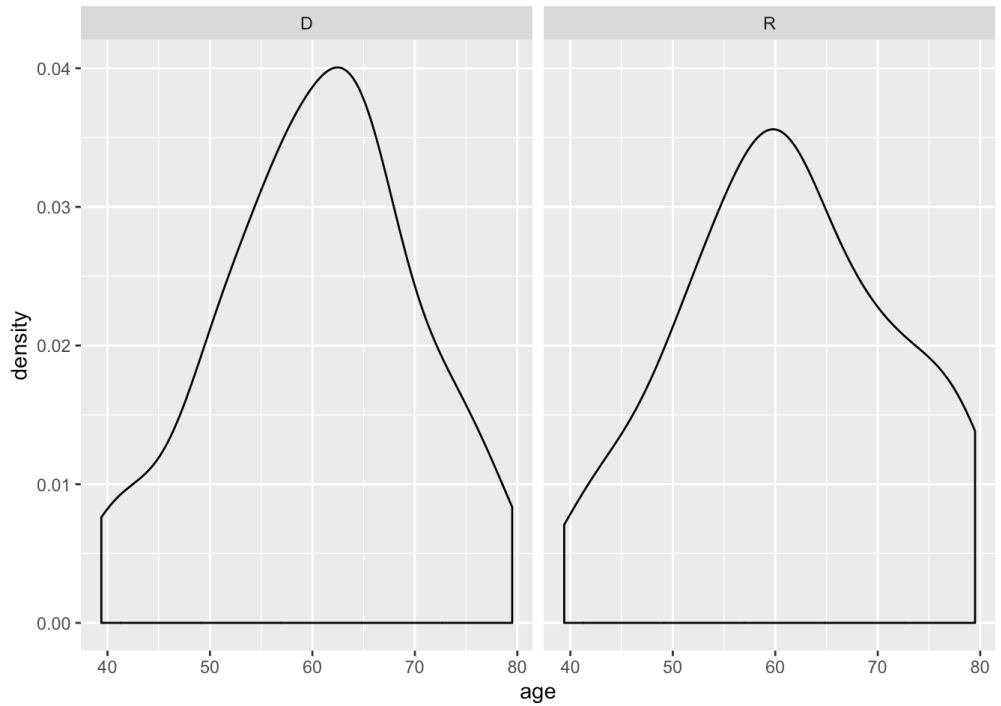
Plot:



Code:

```
ggplot(data = senate_113, mapping = aes(x = age)) +
    geom_density() +
    facet_wrap(~ party)
```

Plot:



Outline

1. Today: Comparing numerical variables across groups

2. Main ideas
    1. Visualization with ggplot
    2. Data wrangling

`https://en.wikipedia.org/wiki/The_Treachery_of_Images`

Rene Magritte, 1929

dplyr functions work with pipes and expect **tidy data**. In tidy data:

&

Each **variable** is in its own **column**

Each **observation**, or **case**, is in its own **row**

pipes

x %>% f(y) becomes **f(x, y)**

Slides from Ben Baumer

The expression

```
mydata %>%
verb(arguments)
```

is the same as

```
verb(mydata, arguments)
```

Thus,

```
function(x, args)
```

has the same effect as

```
x %>%
function(args)
```

Slides from Ben Baumer

Instead of having to read/write:

```
select(filter(mutate(data, args1), args2), args3)
```

You can write:

```
data %>%
    mutate(args1) %>%
    filter(args2) %>%
    select(args3)
```

Slides from Ben Baumer

https://youtu.be/R6xKM-H2awE

Slides from Ben Baumer

Nested form:

```
bop(scoop(hop(foo_foo, through = forest), up = field_mice), on = head)
```

With pipes:

```
foo_foo %>%
    hop(through = forest) %>%
    scoop(up = field_mouse) %>%
    bop(on = head)
```

Slides from Ben Baumer

Code:

```
senate_113 %>%
    group_by(party) %>%
    summarize(
        mean_wt = mean(age),
        median_wt = median(age),
        q1_wt = quantile(age, probs = 0.25),
        q3_wt = quantile(age, probs = 0.75),
        iqr_wt = IQR(age),
        var_wt = var(age),
        sd_wt = sd(age)
    )
```

Output:

party <chr>	mean_wt <dbl>	median_wt <dbl>	q1_wt <dbl>	q3_wt <dbl>	iqr_wt <dbl>	var_wt <dbl>	sd_wt <dbl>
D	60.38679	61.6	54.2	66.0	11.8	94.52694	9.722496
R	61.20000	60.9	53.3	69.1	15.8	110.61773	10.517496

Code:

```
senate_113 ← senate_113 %>%
    arrange(age)
head(senate_113)
```

Output:

firstname <chr>	middlename <chr>	lastname <chr>	birthday <date>	state <chr>	party <chr>	age <dbl>
Christopher	S.	Murphy	1973-08-03	CT	D	39.4
Brian	Emanuel	Schatz	1972-10-20	HI	D	40.2
Martin	*NA*	Heinrich	1971-10-17	NM	D	41.2
Marco	*NA*	Rubio	1971-05-28	FL	R	41.6
Mike	*NA*	Lee	1971-06-04	UT	R	41.6
Ted	*NA*	Cruz	1970-12-22	TX	R	42.0

Code:

```
senate_113 ← senate_113 %>%
    arrange(desc(age))
head(senate_113)
```

Output:

firstname <chr>	**middlename** <chr>	**lastname** <chr>	**birthday** <date>	**state** <chr>	**party** <chr>	**age** <dbl>
Dianne	*NA*	Feinstein	1933-06-22	CA	D	79.5
Charles	E.	Grassley	1933-09-17	IA	R	79.3
Orrin	G.	Hatch	1934-03-22	UT	R	78.8
Richard	C.	Shelby	1934-05-06	AL	R	78.7
Carl	*NA*	Levin	1934-06-28	MI	D	78.5
James	M.	Inhofe	1934-11-17	OK	R	78.1

Code:

```
senate_113 ← senate_113 %>%
      mutate(age = floor(age))
head(senate_113)
```

Output:

firstname <chr>	**middlename** <chr>	**lastname** <chr>	**birthday** <date>	**state** <chr>	**party** <chr>	**age** <dbl>
Dianne	*NA*	Feinstein	1933-06-22	CA	D	79
Charles	E.	Grassley	1933-09-17	IA	R	79
Orrin	G.	Hatch	1934-03-22	UT	R	78
Richard	C.	Shelby	1934-05-06	AL	R	78
Carl	*NA*	Levin	1934-06-28	MI	D	78
James	M.	Inhofe	1934-11-17	OK	R	78