INTRODUCTION TO DATA

Welcome to the course!

High School and Beyond

id	gender	race	...	socst
70	male	white	...	57
121	female	white	...	61
86	male	white	...	31
...	...	...	...	...
137	female	white	...	61

Loading data

```
> # Load package
> library(openintro)

> # Load data
> data(hsb2)
```

Structure of your data

```
> # View the structure of your data
> str(hsb2)
'data.frame': 200 obs. of  11 variables:
 $ id     : int  70 121 86 141 172 113 50 11 84 48 ...
 $ gender : chr  "male" "female" "male" "male" ...
 $ race   : chr  "white" "white" "white" "white" ...
 $ ses    : Factor w/ 3 levels "low","middle",..: 1 2 3 3 2 2 2 2 2 2 ...
 $ schtyp : Factor w/ 2 levels "public","private": 1 1 1 1 1 1 1 1 1 1 ...
 $ prog   : Factor w/ 3 levels "general","academic",..: 1 3 1 3 2 2 1 2 1 2 ...
 $ read   : int  57 68 44 63 47 44 50 34 63 57 ...
 $ write  : int  52 59 33 44 52 52 59 46 57 55 ...
 $ math   : int  41 53 54 47 57 51 42 45 54 52 ...
 $ science: int  47 63 58 53 53 63 53 39 58 50 ...
 $ socst  : int  57 61 31 56 61 61 61 36 51 51 ...
```

Glimpse of your data

```
> # Load package
> library(dplyr)

> # View the structure of your data
> glimpse(hsb2)
Observations: 200
Variables: 11
$ id      <int> 70, 121, 86, 141, 172, 113, 50, 11, 84, 4...
$ gender  <chr> "male", "female", "male", "male", "male",...
$ race    <chr> "white", "white", "white", "white", "whit...
$ ses     <fctr> low, middle, high, high, middle, middle,...
$ schtyp  <fctr> public, public, public, public, public, ...
$ prog    <fctr> general, vocational, general, vocational...
$ read    <int> 57, 68, 44, 63, 47, 44, 50, 34, 63, 57, 6...
$ write   <int> 52, 59, 33, 44, 52, 52, 59, 46, 57, 55, 4...
$ math    <int> 41, 53, 54, 47, 57, 51, 42, 45, 54, 52, 5...
$ science <int> 47, 63, 58, 53, 53, 63, 53, 39, 58, 50, 5...
$ socst   <int> 57, 61, 31, 56, 61, 61, 61, 36, 51, 51, 6...
```

INTRODUCTION TO DATA

Let’s practice!

INTRODUCTION TO DATA

Types of variables

Types of variables

- **Numerical (quantitative):** numerical values
  - **Continuous:** infinite number of values within a given range, often measured
  - **Discrete:** specific set of numeric values that can be counted or enumerated, often counted
- **Categorical (qualitative):** limited number of distinct categories
  - **Ordinal:** finite number of values within a given range, often measured

Glimpse to identify variables

```
> # Load package
> library(dplyr)

> # View the structure of your data
> glimpse(hsb2)
Observations: 200
Variables: 11
$ id      <int> 70, 121, 86, 141, 172, 113, 50, 11, 84, 4...
$ gender  <chr> "male", "female", "male", "male", "male",...
$ race    <chr> "white", "white", "white", "white", "whit...
$ ses     <fctr> low, middle, high, high, middle, middle,...
$ schtyp  <fctr> public, public, public, public, public, ...
$ prog    <fctr> general, vocational, general, vocational...
$ read    <int> 57, 68, 44, 63, 47, 44, 50, 34, 63, 57, 6...
$ write   <int> 52, 59, 33, 44, 52, 52, 59, 46, 57, 55, 4...
$ math    <int> 41, 53, 54, 47, 57, 51, 42, 45, 54, 52, 5...
$ science <int> 47, 63, 58, 53, 53, 63, 53, 39, 58, 50, 5...
$ socst   <int> 57, 61, 31, 56, 61, 61, 61, 36, 51, 51, 6...
```

Let’s practice!

INTRODUCTION TO DATA

Categorical data in R: factors

Categorical data

- Often stored as factors in R
  - Important use: statistical modeling
  - Sometimes undesirable, sometimes essential
- Common in subgroup analysis
  - Only interested in a subset of the data
  - Filter for specific levels of categorical variable

Table to explore

```
> # Number of students in public and private schools in hsb2
> table(hsb2$schtyp)

 public private 
    168      32 
```

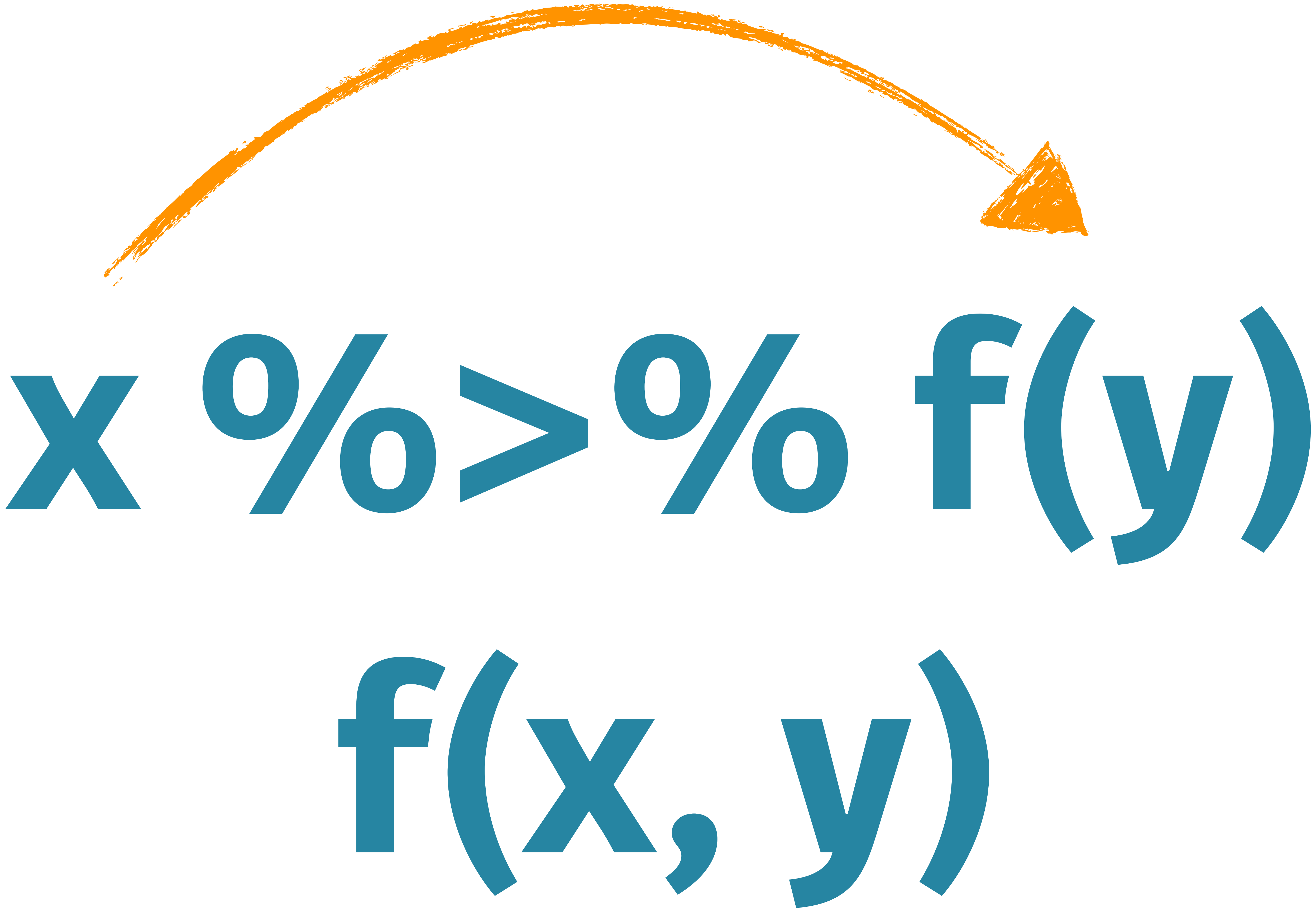
Filter to subset

```
> # Filter for public schools
> hsb2_public <- hsb2 %>%
    filter(schtyp == "public")
```

The pipe operator

%>%

The pipe operator



x %>% f(y)

f(x, y)

A (very) simple pipe

```
> # Sum of 3 and 4, without pipe
> sum(3, 4)
[1] 7
```

```
> # Sum of 3 and 4, with pipe
> 3 %>% sum(4)
[1] 7
```

Filter to subset (cont.)

```
> # Filter for public schools
> hsb2_public <- hsb2 %>%
    filter(schtyp == "public")
```

"is equal to"

Table to explore further

```
> # Number of students in public and private schools in hsb2_public
> table(hsb2_public$schtyp)

 public private
    168       0
```

Drop (unused) levels

```
> # Drop unused levels
> hsb2_public$schtyp <- droplevels(hsb2_public$schtyp)
```

```
> # Number of students in public and private schools in hsb2_public
> table(hsb2_public$schtyp)

public
   168
```

Let’s practice!

Discretize a variable

Average reading score

```
> # Calculate average reading score and show the value
> mean(hsb2$read)
[1] 52.23

> # Calculate average reading score and store as avg_read
> avg_read <- mean(hsb2$read)

> # Do both
> (avg_read <- mean(hsb2$read))
[1] 52.23
```

New variable: read_cat

id	...	read	read_cat
70	...	57	at or above avg
121	...	68	at or above avg
86	...	44	below avg
...	...	...	...
137	...	63	at or above avg

New variable: read_cat

```
> # Create new variable: read_cat
> hsb2 <- hsb2 %>%
    mutate(read_cat = ifelse(read < avg_read,
           "below average", "at or above average"))
```

logical test

if TRUE

if FALSE

Let’s practice!

Visualizing numerical data

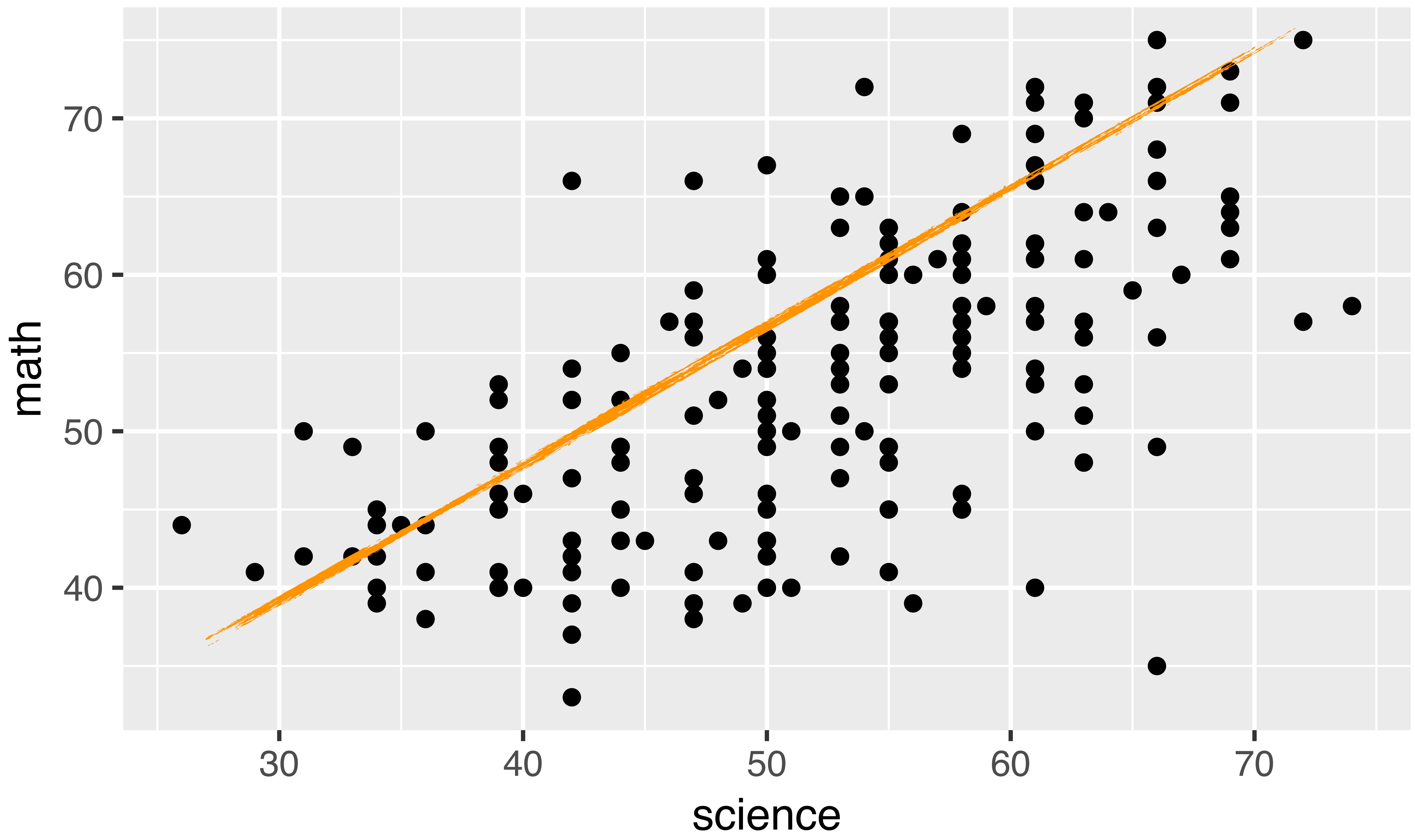
ggplot2

- Modern looking, hassle-free plots
- Easy to extend code for multivariate plots
- Iterative construction

```
> # Load ggplot2
> library(ggplot2)
```

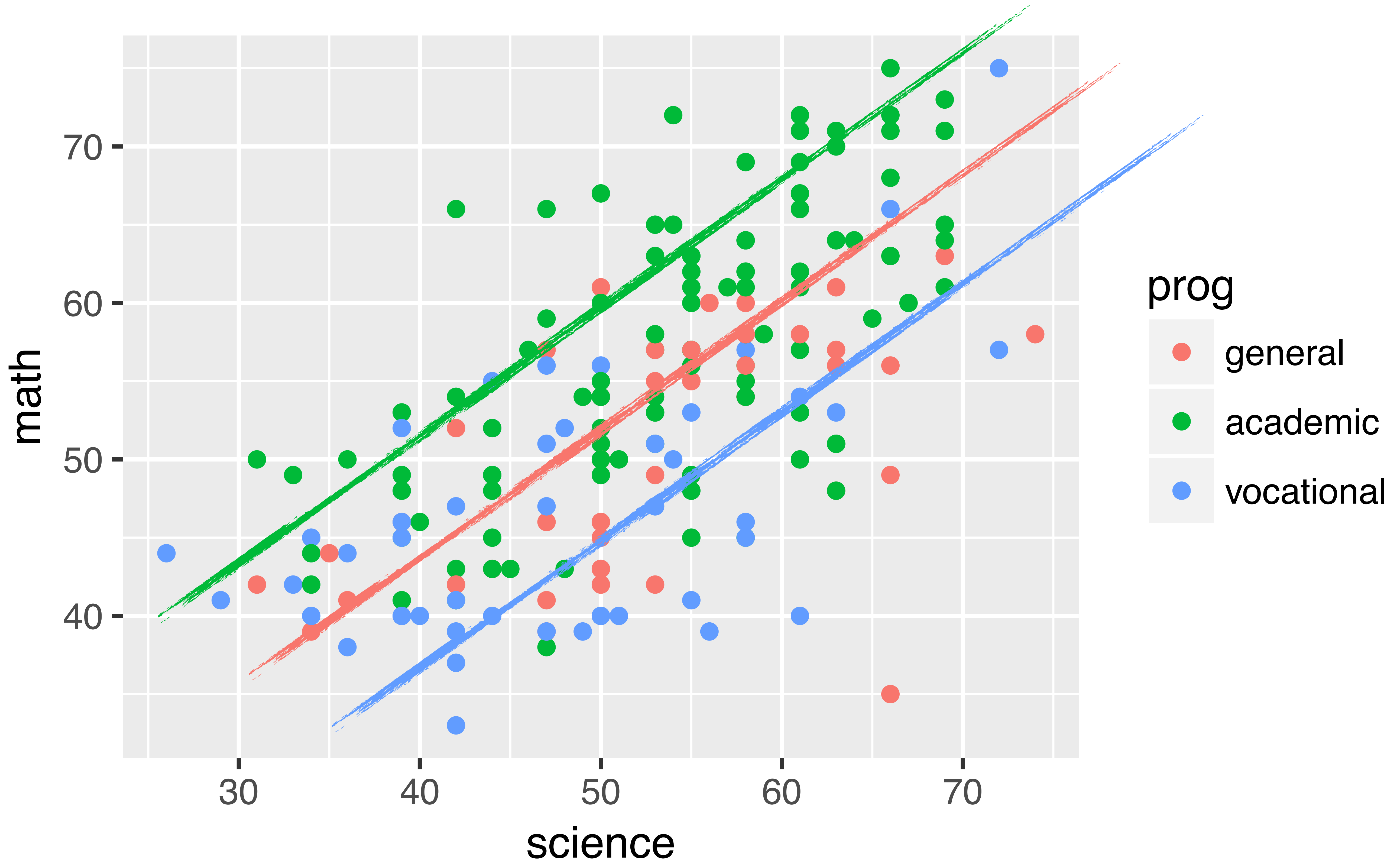
Math vs. science scores

```
> # Scatterplot of math vs. science scores
> ggplot(data = hsb2, aes(x = science, y = math)) +
    geom_point()
```



Math, science, and program

```
> # Scatterplot of math vs. science scores, controlling for program
> ggplot(data = hsb2, aes(x = science, y = math, color = prog)) +
    geom_point()
```



Let’s practice!