Today

- Recap & questions from homework
- Coding the grid search algorithm

Structured programming

- Sequence structure
- Selection structure (conditional, branches)
- Repetition structure (iteration, loops)

R selection structures

if single selection structure

```
if ( condition ) {
    expression
}
```

if-else double selection structure

```
if ( condition ) {
    expression1
} else {
    expression2
}
```

if-else if multiple selection structure

```
if ( condition ) {
    expression1
} else if {
    expression2
} else {
    expression3
}
```

R repetition structures in practice

while sentinel control

```
while ( condition ) {
   expression
}
```

for counter control

```
for ( i in 1:n ) {
    expression
}
```

until sentinel control

```
while ( !condition ) {
   expression
}
```

foreach vector control

```
for ( element in vector ) {
    expression
}
```

do-while sentinel control (e.g. option 4)

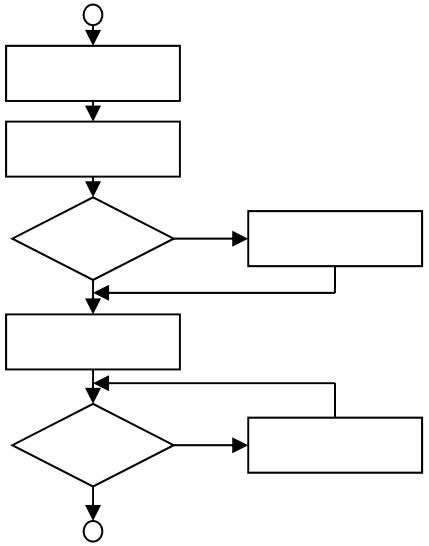
```
repeat {
    expression
    if ( !condition ) break
}
```

foreach vector control with lists

```
for ( object in list ) {
    expression
}
```

Combining control structures

- Stacking
  - one after another
- Nesting
  - one inside another



These are all the programming tools you need to solve any problem!

Next: additional, powerful programming tools for convenience or to solve specific problems.

Algorithms in data science

- Model algorithm
- Training algorithm
- Inference algorithm

How can a model be an algorithm?

Model

$$y_i = \beta_0 + \beta_1 x_i$$

Algorithm (this version is atomic code)

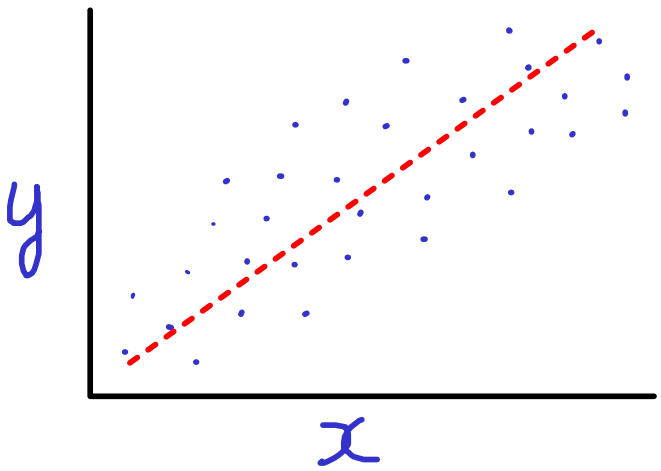
```
> for ( i in 1:n ) {
>   y[i] = b_0 + b_1 * x[i]
> }
```

Vectorized R code

```
> y <- b_0 + b_1 * x
```

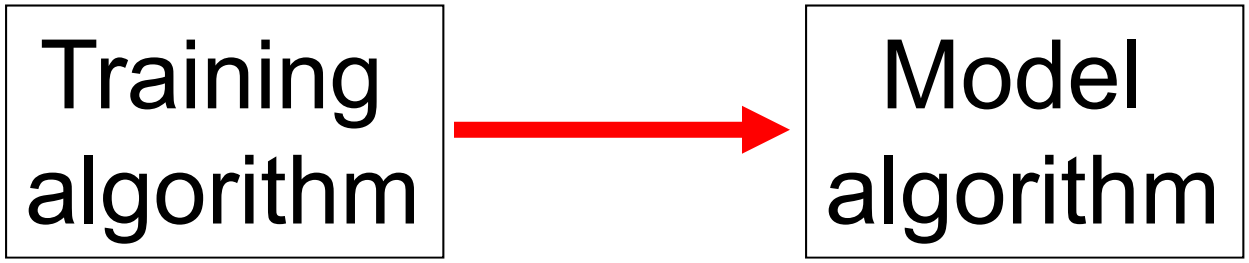
Data table

i	y	x
1	28.4	10.2
2	47.6	15.7
...		
...		
85	35.1	12.9



Training algorithm

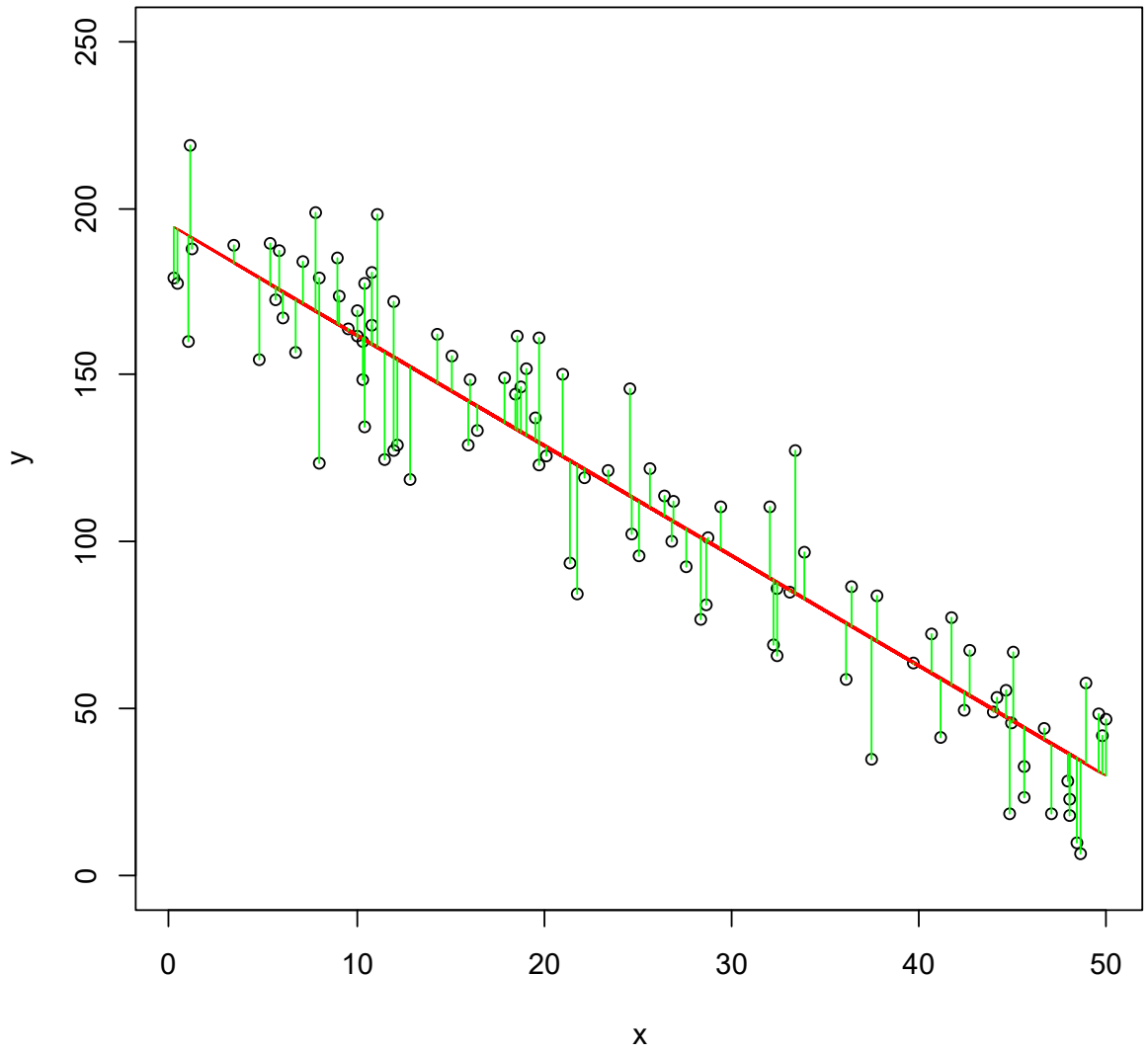
aka model fitting, model calibration



Big idea in data science

Legendre 1805: comet orbits, SSQ.

Least squares training algorithm



Vary model parameters

$$y_i = \beta_0 + \beta_1 x_i + e_i$$

Minimize distance from data

$$ssq = \sum_i e_i^2$$

Optimization algorithms

- Grid search
- Descent
- Monte Carlo
- Analytical or numerical

Grid search algorithm Pseudocode

Read in data
Set up values of β_0 and β_1 to try
Set up storage for ssq, β_0, β_1
For each value of β_0
    For each value of β_1
        Calculate sum of squares
        Store ssq, β_0, β_1
Plot sum of squares profiles (ssq vs β_0, ssq vs β_1)
Report best ssq, β_0, β_1
Plot fitted model with the data

Initialization phase

Calculation phase

Termination phase

Grid search algorithm Pseudocode

Read in data
Set up values of β_0 and β_1 to try
Set up storage for ssq, β_0, β_1
For each value of β_0
    For each value of β_1
        Calculate model predictions
        Calculate deviations
        Sum squared deviations
        Store ssq, β_0, β_1
Plot sum of squares profiles (ssq vs β_0, ssq vs β_1)
Report best ssq, β_0, β_1
Plot fitted model with the data

Top down refinement

Grid search algorithm Pseudocode

Read in data
Set up values of β_0 and β_1 to try
Set up storage for ssq, β_0, β_1
For each value of β_0
    For each value of β_1
        Calculate model predictions
        Calculate deviations
        Sum squared deviations
        Store ssq, β_0, β_1
Plot sum of squares profiles (ssq vs β_0, ssq vs β_1)
Report best ssq, β_0, β_1
Plot fitted model with the data

Translate this to R code and use it to train the model with your data