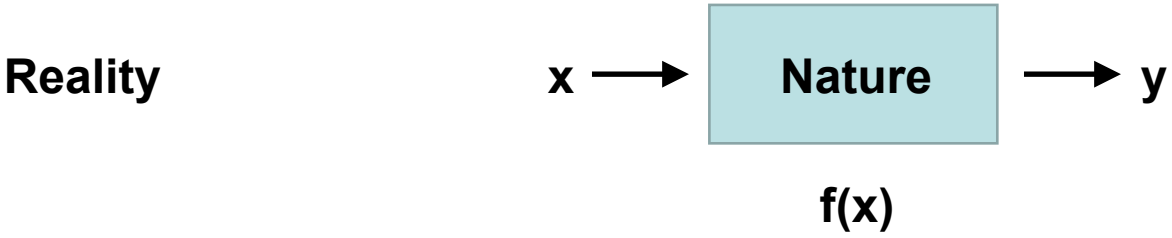
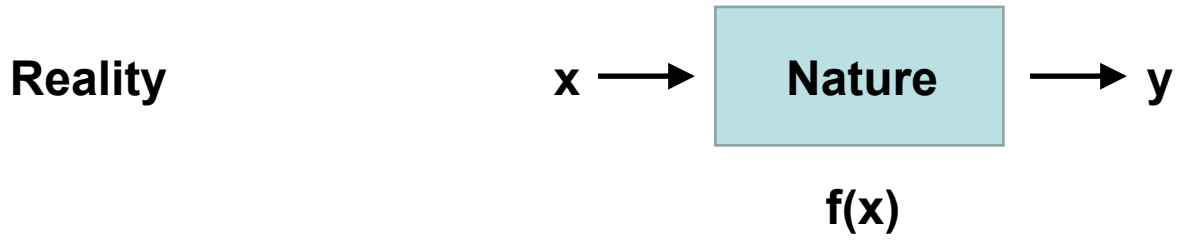
Reminders

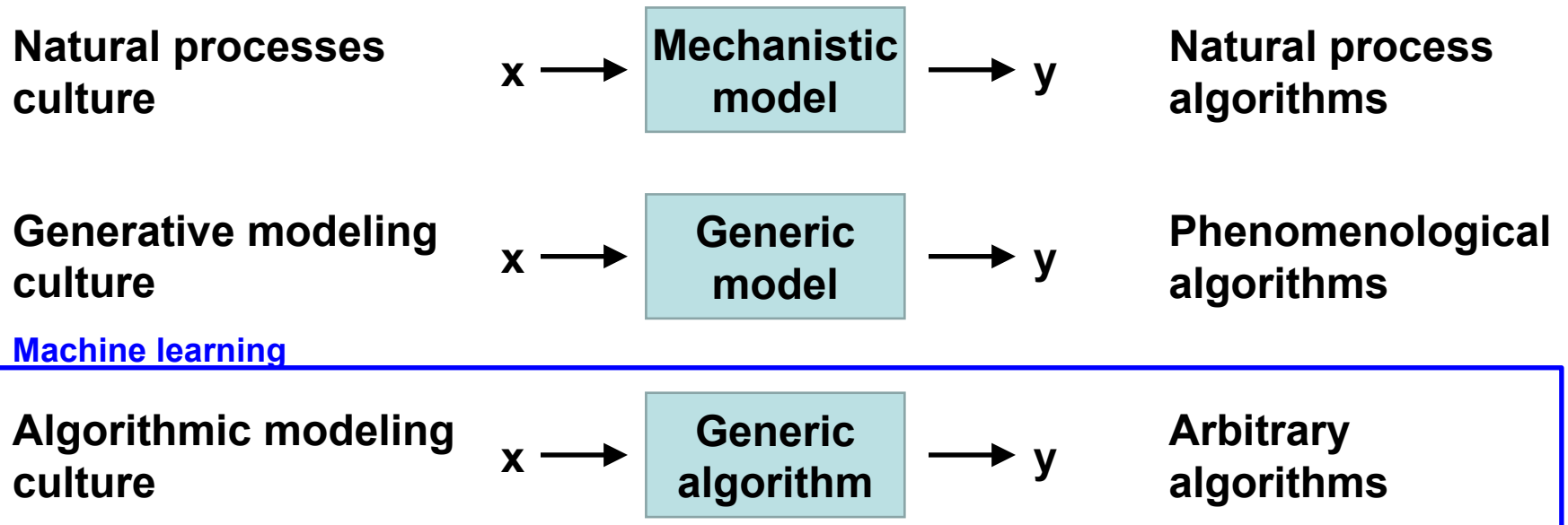
- Take charge of your GitHub repo
- `git clone ...`

Trying to learn a function f



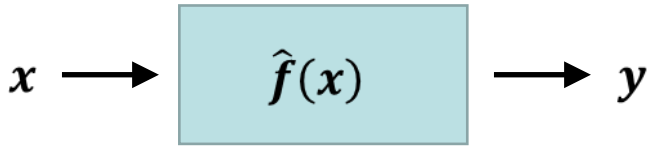
Trying to learn a function f





f can mean different things in different cultures

Prediction is the goal of ML



Goal: find function $\hat{f}$ that has good predictive performance

Accurate on new observations of y

(out-of-sample accuracy)

Machine Learning algorithms

- Model algorithm
- Training algorithm
- Inference algorithm

This week

- Full machine learning workflow
- Machine learning with ants data
  - polynomial model algorithm (nb pedagogical)
  - least squares training algorithm
  - cross validation inference algorithm
- R & Python code

Machine learning workflow

Overall algorithm:

1. Create model algorithm(s) for $\hat{f}(x)$
2. Use a training algorithm to find parameter values of $\hat{f}(x)$
3. Use an inference algorithm to measure prediction error and compare predictive skill among models (model families, tuning parameters, x sets, etc).

Basic full ML setup

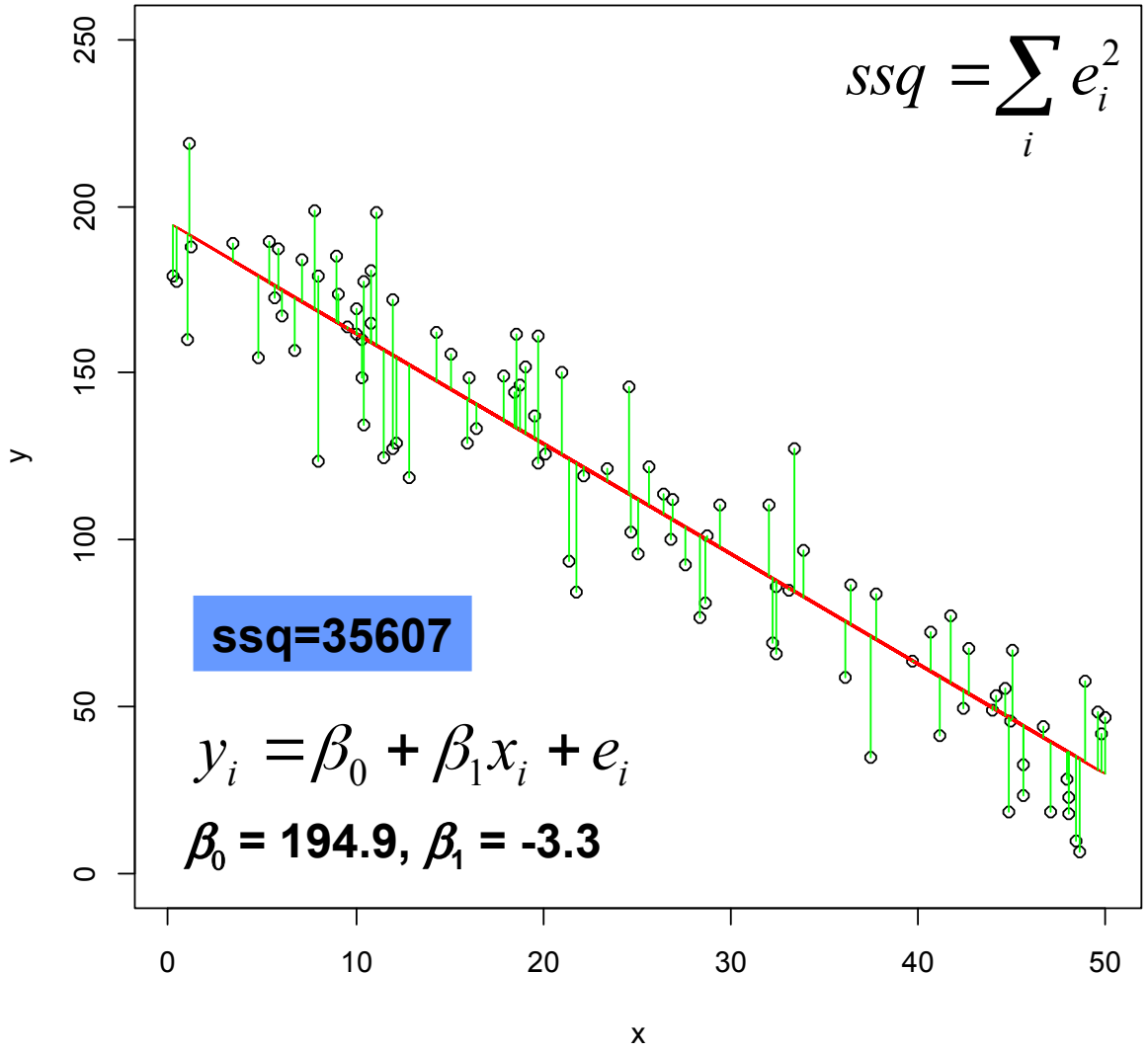
- 3 algorithms:
  - model: flexible function $\hat{f}(x)$;
    e.g. polynomial linear model

    $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \ldots + \beta_m x^m$ m=order
  - training: optimize objective function
    e.g. least squares
  - minimize $SSQ = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$ for training data
  - inference: measure error by cross validation; tuning parameter (order of poly)

Code

- 02_3_ants_cv_polynomial.R
- 02_3_ants_cv_polynomial.py

Least squares optimization



General algorithmic idea:

Vary model parameters until we find the parameter values that minimize the distance of the model from the data

Optimization algorithms

Strategies

1. Systematically try all combinations of parameters - Grid search algorithms
2. Narrowing in: keep changing parameters in the direction that leads to lower SSQ - Descent algorithms
3. Try random values for parameter combinations - Monte Carlo algorithms
4. Solve for parameters using math - Analytical or numerical algorithms

Linear regression in R uses strategy 4

lm(y ~ x) solves a system of linear equations using linear algebra

Mathematical theory shows what to do (QR decomposition)

Numerical algorithm is needed to do it (householder algorithm)

Fast, specialized, guaranteed to find the minimum SSQ. Only works for SSQ: **limited to ordinary linear regression**.