A model is an abstraction, simplification, or representation of a real system

Forms of Models

I) Physical   vs.   Abstract

Airplane in
wind tunnel

Non-physical
representation

Tube with pins

- Diagrams
- Conceptual
  (verbal)
- Mathematical
  model

Represents a real system
with 1 or more equations

II) Empirical vs. Mechanistic

- Collect experimental data
- Use statistical methods
- Intended as <u>predictive</u>
- Interested in *how* system acts rather than *why*
- Experiment first, model later
- Parameters – unclear biological meaning

- Based on the structure of the system
- Used to gain an understanding of mechanisms
- Often <u>reductionist</u>
- Model first, experiment later
- Hypothesis generating
- Parameters – clear biological meaning

Reductionist – Break down system into little parts, study parts and then describe the system as a combination of those parts

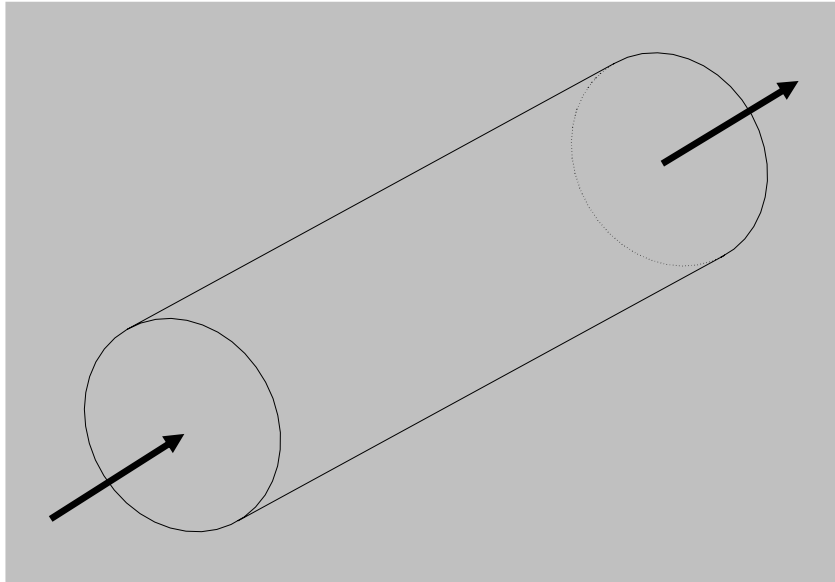# Empirical model – Rainfall during the year
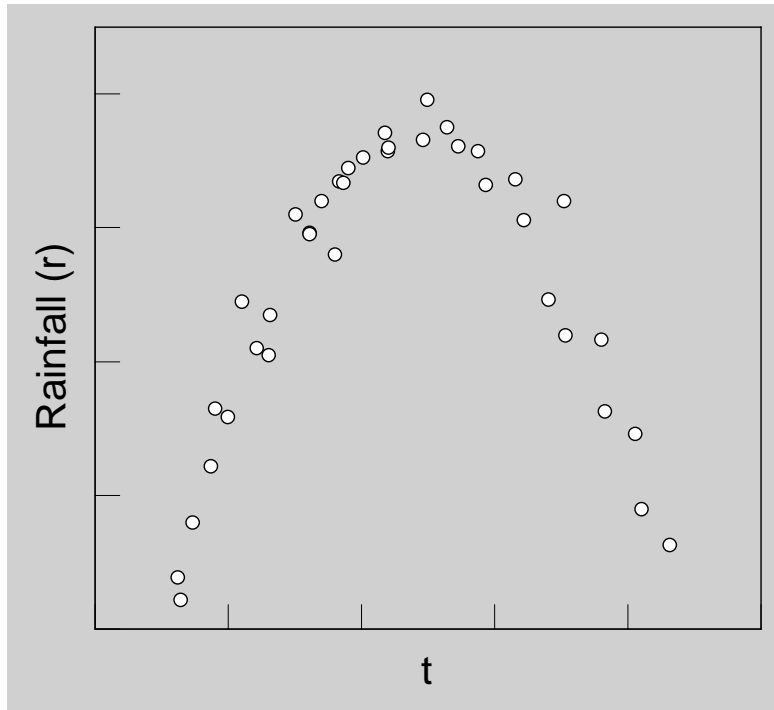
Want a predictive tool!

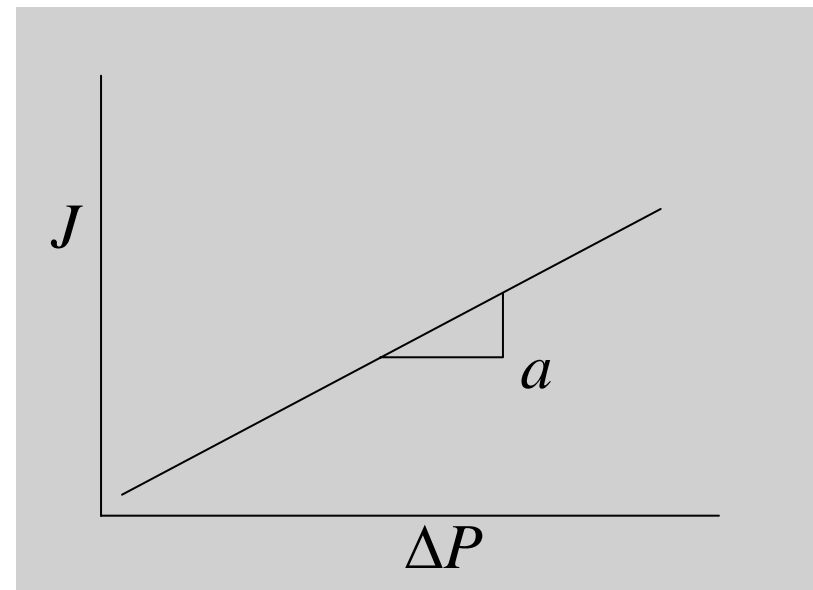Use a statistical fit to a parabola:

$$r = at^2 + bt + c$$

Parameters

Parameter meaning is not clear - don't care, want predictive tool!!

Mechanistic model – Water flow thru a pipe

$$J = a \, \Delta P$$

Interested in mechanisms
What is "a"?
Use theory:

$$a = \frac{\pi r^4}{viscosity}$$

"a" has biological meaning –
related to the radius of the pipe!

III)   Static   vs.   Dynamic

- Time not explicitly considered
- Steady-state models
- Flows are constant
- Pool sizes are constant

- Does not mean nothing moves!

Filling can example…

- Consider changes with time
- Transient-state models
- Changing flows or pool sizes

Model Uses

- Predictive
- Theoretical
- Synthesis
- Communication

Model Solutions (mathematical models)

A) Analytical

Symbolic manipulation of equations

$$y = x^2$$, want to know x

$$x = \sqrt{y}$$

B) Numerical

Equations not manipulated symbolically
Solution provided as numerical data

$$y = x^2$$

What is x if $y = 5$?
Repeated guessing, interval halving

Some definitions…

<u>Deterministic</u> : current state of a system is determined by previous states – future states can be predicted (*vs*. random)

<u>Robust</u> : Conclusions from a model are relatively insensitive to model structure

<u>Realistic</u> : model accounts for all relevant variables and processes (difficult to know)

Programming for Modeling

Computer operating system -

Program that :

- Loads, starts programs
- Helps programs communicate with
  - Display
  - Keyboard (input devices)
  - Ports
  - Disk drives (storage)

Operating system manages programs and these devices

Program – series of instructions to direct computer on processing information

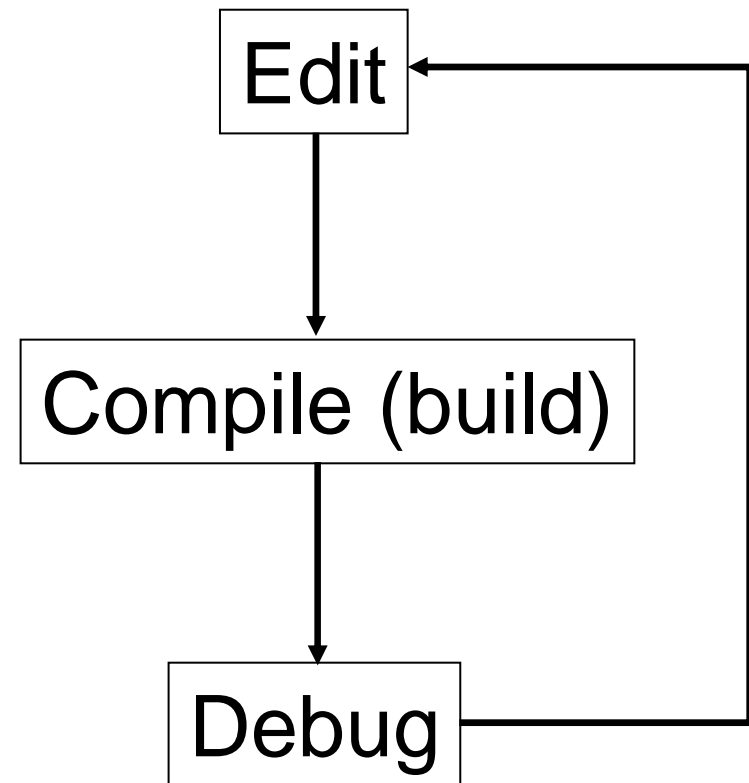Programs written in a programming language (readable by us!)

(BASIC, Pascal, C, Fortran, Java)

Must be converted to machine language (readable by computer)

Compiler – program that translates

Program writing – often done with help
of software that combines :

- Editor

- Compiler

- Debugger

Edit → Compile (build) → Debug → (back to Edit)

Notes from Course Syllabus

- It is ok and sometimes useful to work with other students in the development of your programs. BUT, you must turn in your own program and project reports and not some kind of shared copy of program, results, and report.

- Reviewing the programs of other students along with the sample programs available on the course web page can help you to learn programming and to develop the programs for our projects, but it will be essential that you <u>understand</u> the programs in terms of how they work and what each statement means.

- Sometimes students are tempted to get other people to write their programs for them – this is generally a bad idea, because you will not end up learning anything about writing programs yourself and you will not understand the ones you see in class or are asked about on an exam.

Programming in C

The Icc compiler – see handout

Functions in C – the main function

Functions are blocks of program that carry out a particular task – often useful to breakup task into parts.

All programs have 1 function called "main"

```
int main ( void )
{
        -
        -
     return 0;
}
```

Data variable types

Integers

short	0 - 255 (2^8)	$\pm$127
int *	0 - 65,535	$\pm$32767
long	0 - 4E9	$\pm$ 2E9

Floating point, decimal

float	$\pm10^{\pm38}$	6 digits
double	$\pm10^{\pm308}$	15 digits

Text ?

* - for most desktop computers, the int type is the same as the long type

<u>Array</u> – group of variables with same name

Think of as a table with rows and columns

x[ ] – an array called x (column of numbers)

x[ ] [ ] – a 2D array (rows and columns)

x[0] – number in the first row of x
x[1] – number in the second row
x[2][3] – third row, fourth column

Storing text

char (stores a number from 0 – 255)
Words – strings of text – stored as an array of chars.

Declaring variables – before using a variable, C must know to set aside space.

```
int n;
unsigned int n;
float x;
float x[100];
char a[100];
```