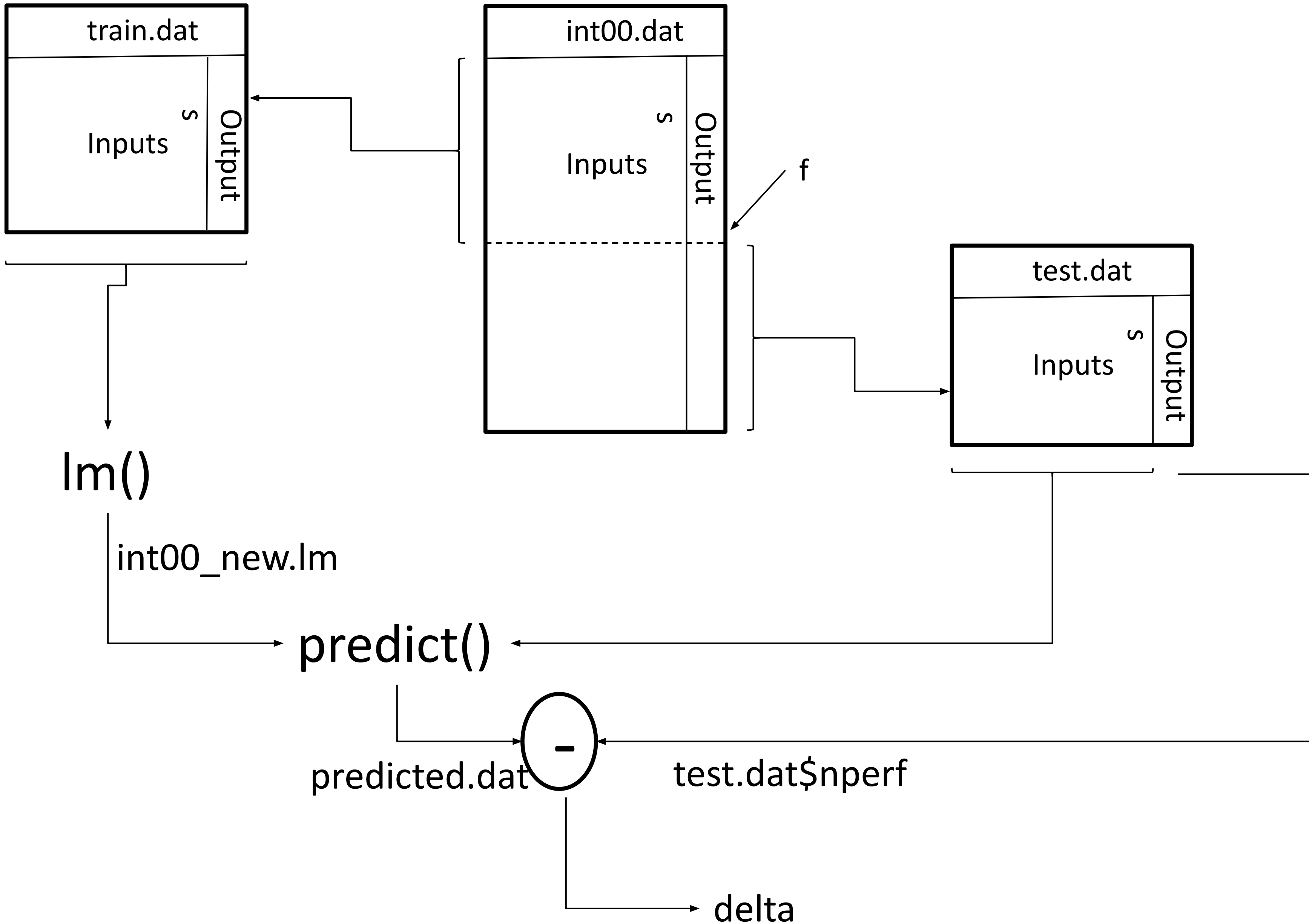
Lab 5: Training, testing, predicting

- Best indicator of a model's quality is its ability to accurately predict output values given previously unseen input values.
- But how to measure this prediction ability?
- Use some of the data to <u>*train*</u> the model
- Then use remainder of the data to <u>*test*</u> the model's predictions

Data splitting



Data splitting

```
# Do NOT set random number seed for labs!
rows <- nrow(int00.dat)
f <- 0.5
upper_bound <- floor(f * rows)
permuted_int00.dat <- int00.dat[sample(rows) , ]
train.dat <- permuted_int00.dat[1:upper_bound, ]
test.dat <- permuted_int00.dat[(upper_bound+1): rows, ]
```

sample(n) □ permutation of integers from 1 to n

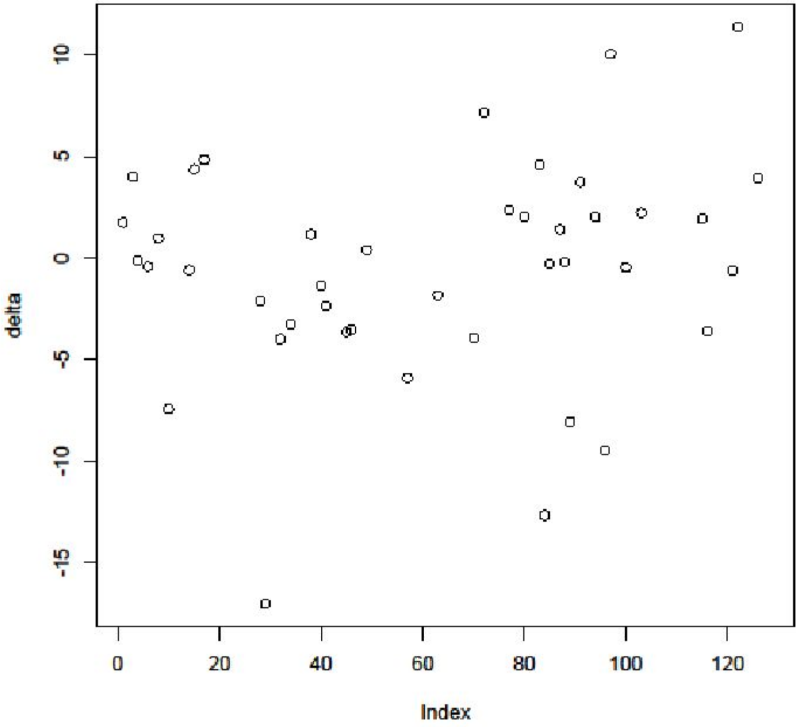
- *New permutation each time sample(n) is called if random seed is not set*

Training and testing

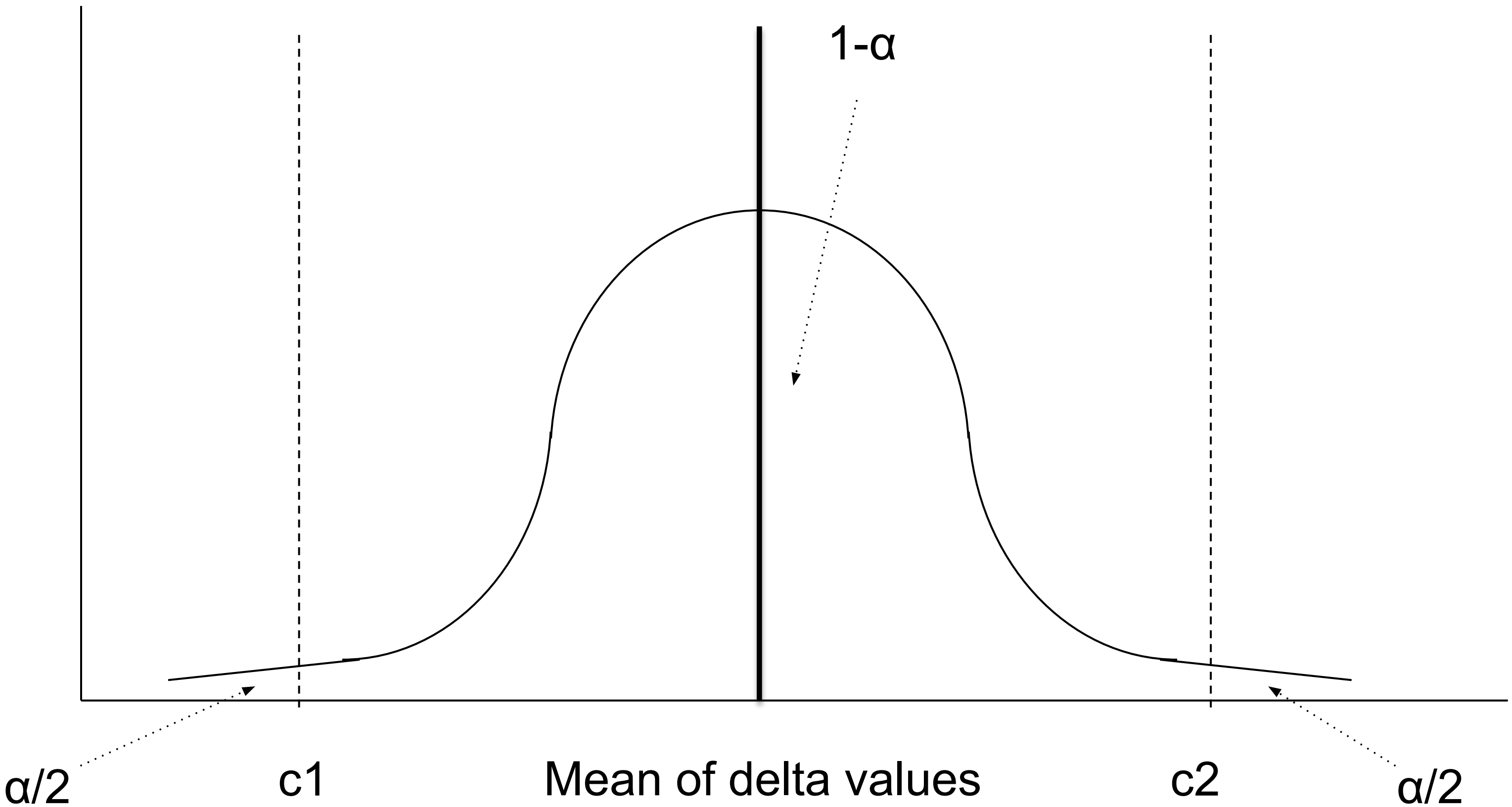
- int00_new.lm <- lm(nperf ~ clock + cores + … , data = train.dat)
- predicted.dat <- predict(int00_new.lm, newdata=test.dat)
- delta <- predicted.dat – test.dat$nperf

Training and testing

- plot(delta)
- Expect values uniformly distributed around 0
- Use t-test to generate confidence interval for the mean of delta.
- Is the average of delta = 0?



Confidence Interval for the Mean



Normalize *x*

$$z = \frac{\bar{x} - x}{s / \sqrt{n}}$$

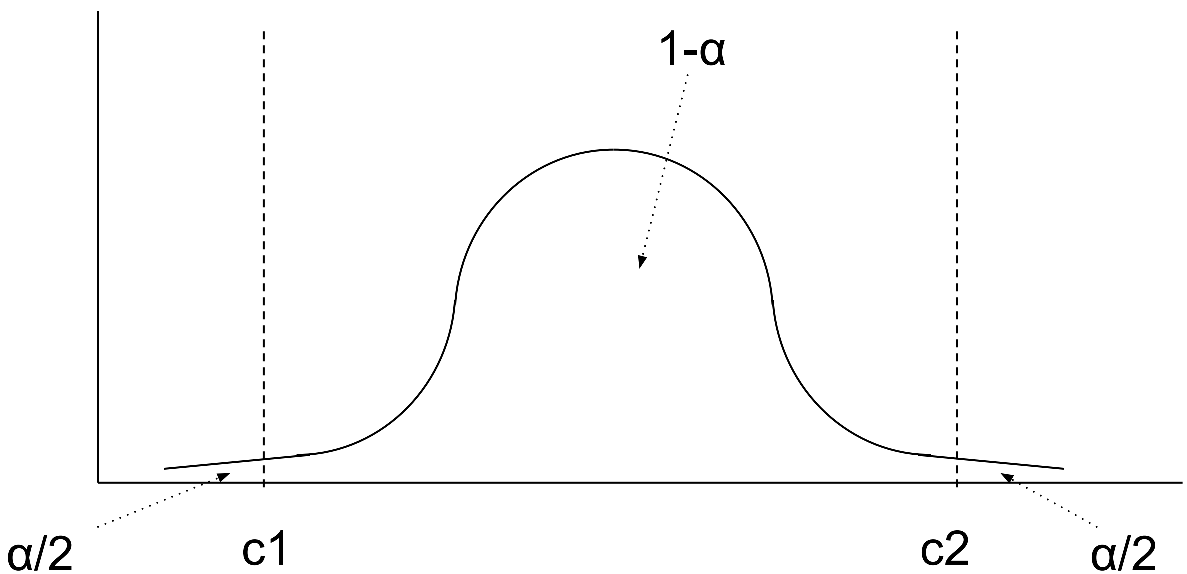
$$n = \text{number of measurements}$$

$$\bar{x} = \text{ mean} = \sum_{i=1}^{n} x_i$$

$$s = \text{standard deviation} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} (x_i - \bar{x})^2}{n-1}$$

Confidence Interval for the Mean

- Normalized *z* follows a Student's *t* distribution
  - (n-1) degrees of freedom
  - Area left of $c_2 = 1 - \alpha/2$
  - Tabulated values for *t*



Confidence Interval for the Mean

From the Student's t-table

$$c_1 = \bar{x} - t_{1-\alpha/2;n-1} \frac{s}{\sqrt{n}}$$

$$c_2 = \bar{x} + t_{1-\alpha/2;n-1} \frac{s}{\sqrt{n}}$$

Then,

$$\Pr(c_1 \le x \le c_2) = 1 - \alpha$$

An Example

Experiment	Measured value
1	8.0 s
2	7.0 s
3	5.0 s
4	9.0 s
5	9.5 s
6	11.3 s
7	5.2 s
8	8.5 s

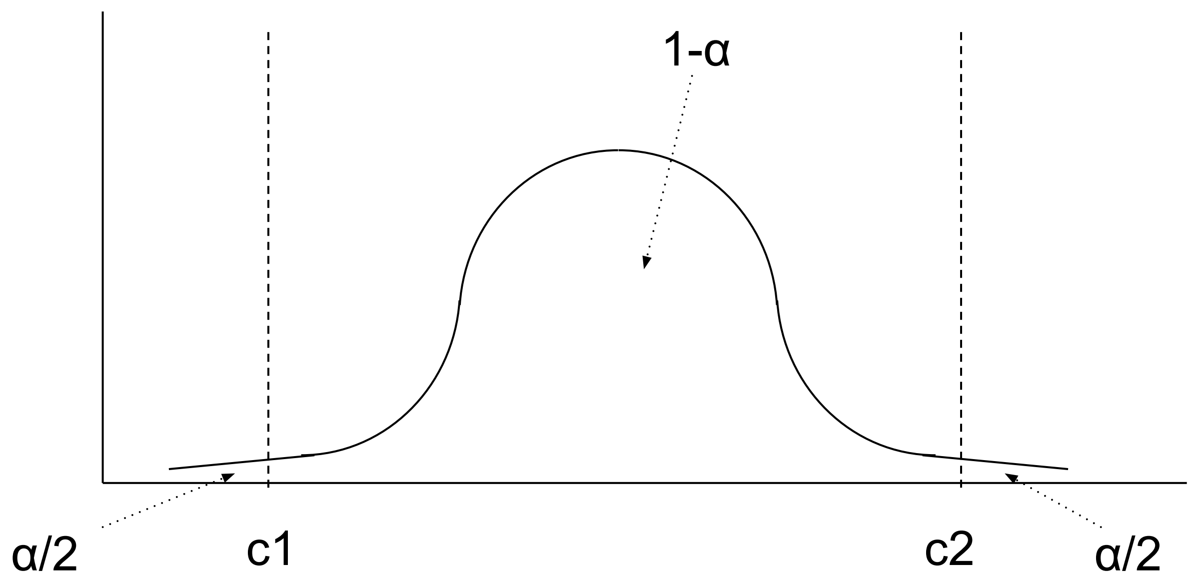
An Example (cont.)

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} = 7.94$$

$$s = \text{sample standard deviation} = 2.14$$

An Example (cont.)

- 90% CI $\longrightarrow$ 90% chance actual value in interval
- 90% CI $\longrightarrow$ $\alpha = 0.10$
  - $1 - \alpha /2 = 0.95$
- $n = 8 \longrightarrow 7$ degrees of freedom



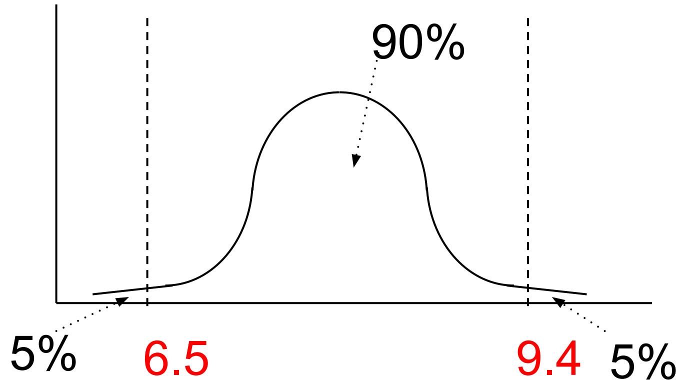
90% Confidence Interval

$$a = 1 - \alpha / 2 = 1 - 0.10 / 2 = 0.95$$

$$t_{a;n-1} = t_{0.95;7} = 1.895$$

$$c_1 = 7.94 - \frac{1.895(2.14)}{\sqrt{8}} = 6.5$$

$$c_2 = 7.94 + \frac{1.895(2.14)}{\sqrt{8}} = 9.4$$



	a		
n	0.90	0.95	0.975
…	…	…	…
5	1.476	2.015	2.571
6	1.440	1.943	2.447
7	1.415	1.895	2.365
…	…	…	…
∞	1.282	1.645	1.960

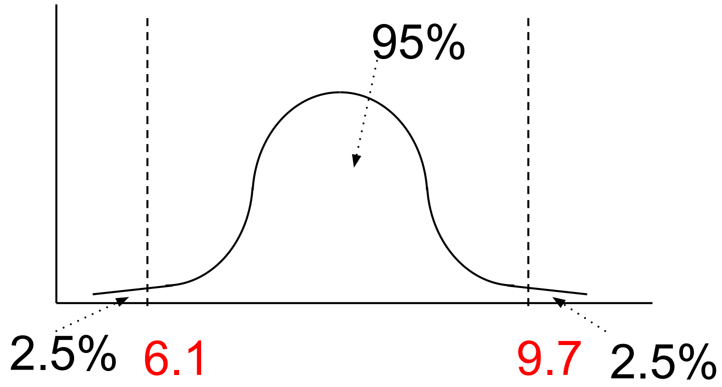
95% Confidence Interval

$$a = 1 - \alpha/2 = 1 - 0.10/2 = 0.975$$

$$t_{a;n-1} = t_{0.975;7} = 2.365$$

$$c_1 = 7.94 - \frac{2.365(2.14)}{\sqrt{8}} = 6.1$$

$$c_2 = 7.94 + \frac{2.365(2.14)}{\sqrt{8}} = 9.7$$

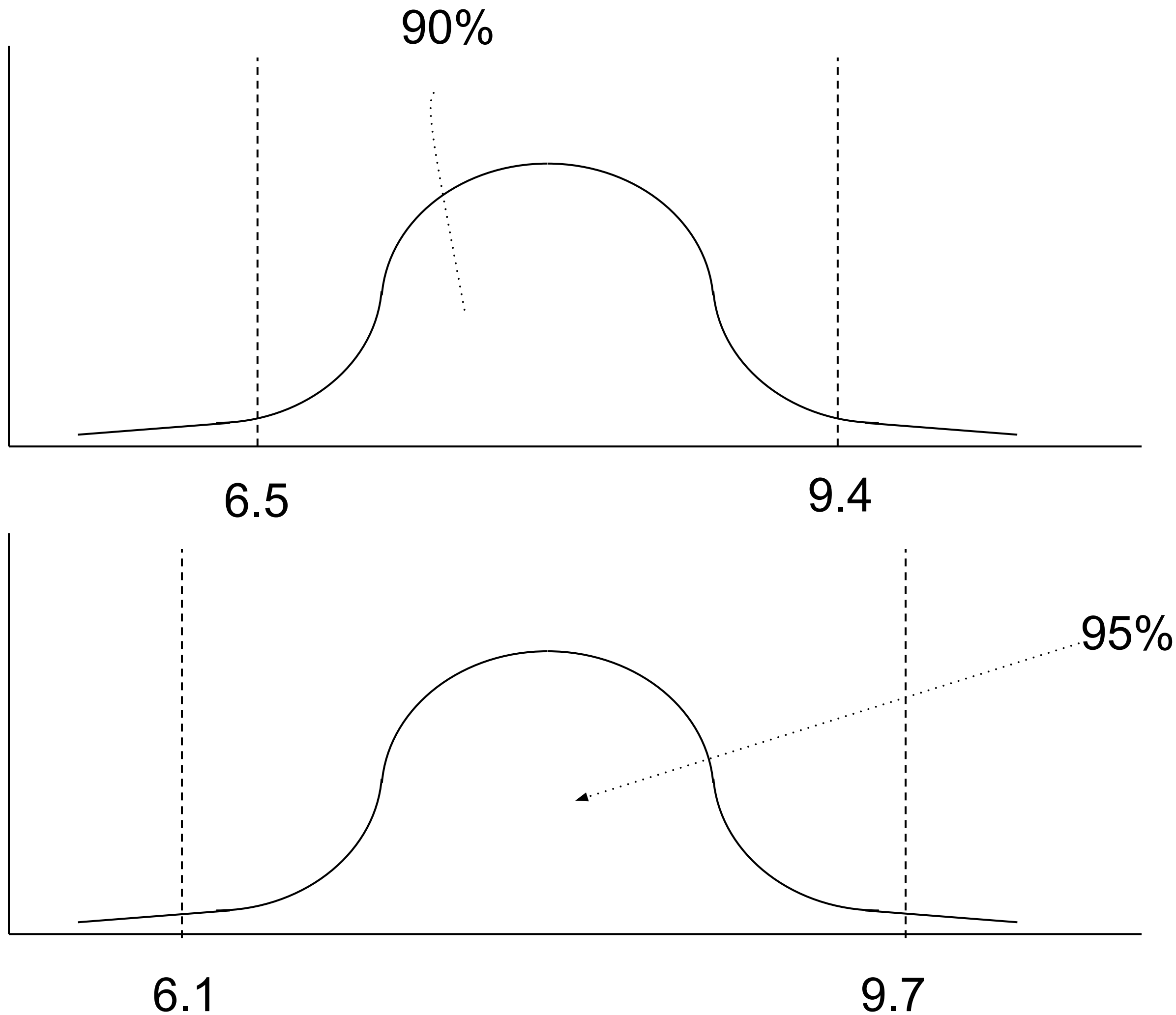


	a		
n	0.90	0.95	0.975
…	…	…	…
5	1.476	2.015	2.571
6	1.440	1.943	2.447
7	1.415	1.895	2.365
…	…	…	…
∞	1.282	1.645	1.960

What does it mean?

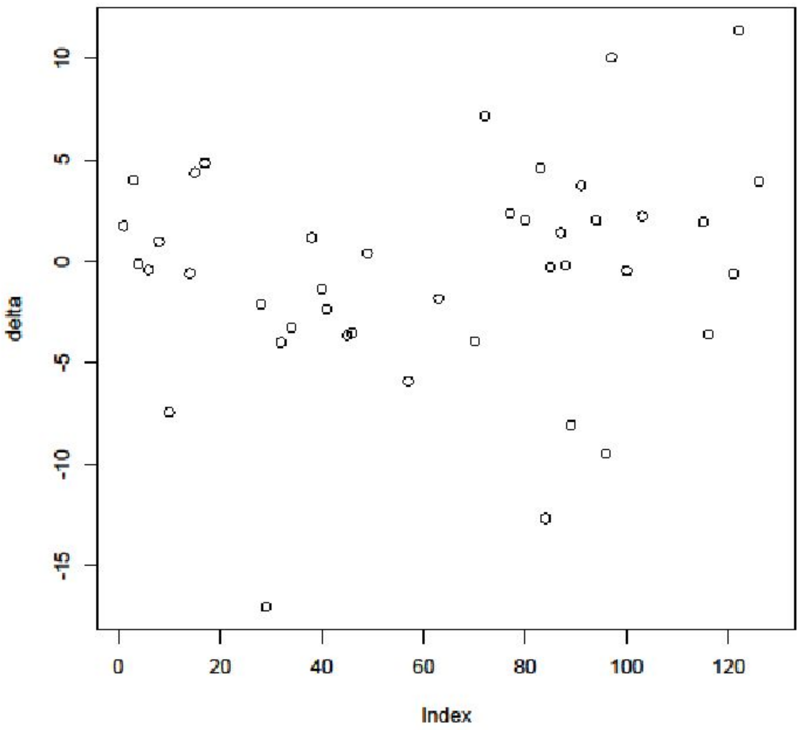
- 90% CI = [6.5, 9.4]
  - 90% chance real value is between 6.5, 9.4
- 95% CI = [6.1, 9.7]
  - 95% chance real value is between 6.1, 9.7
- Why is interval wider when we are more confident?

Higher Confidence → Wider Interval?



Training and testing

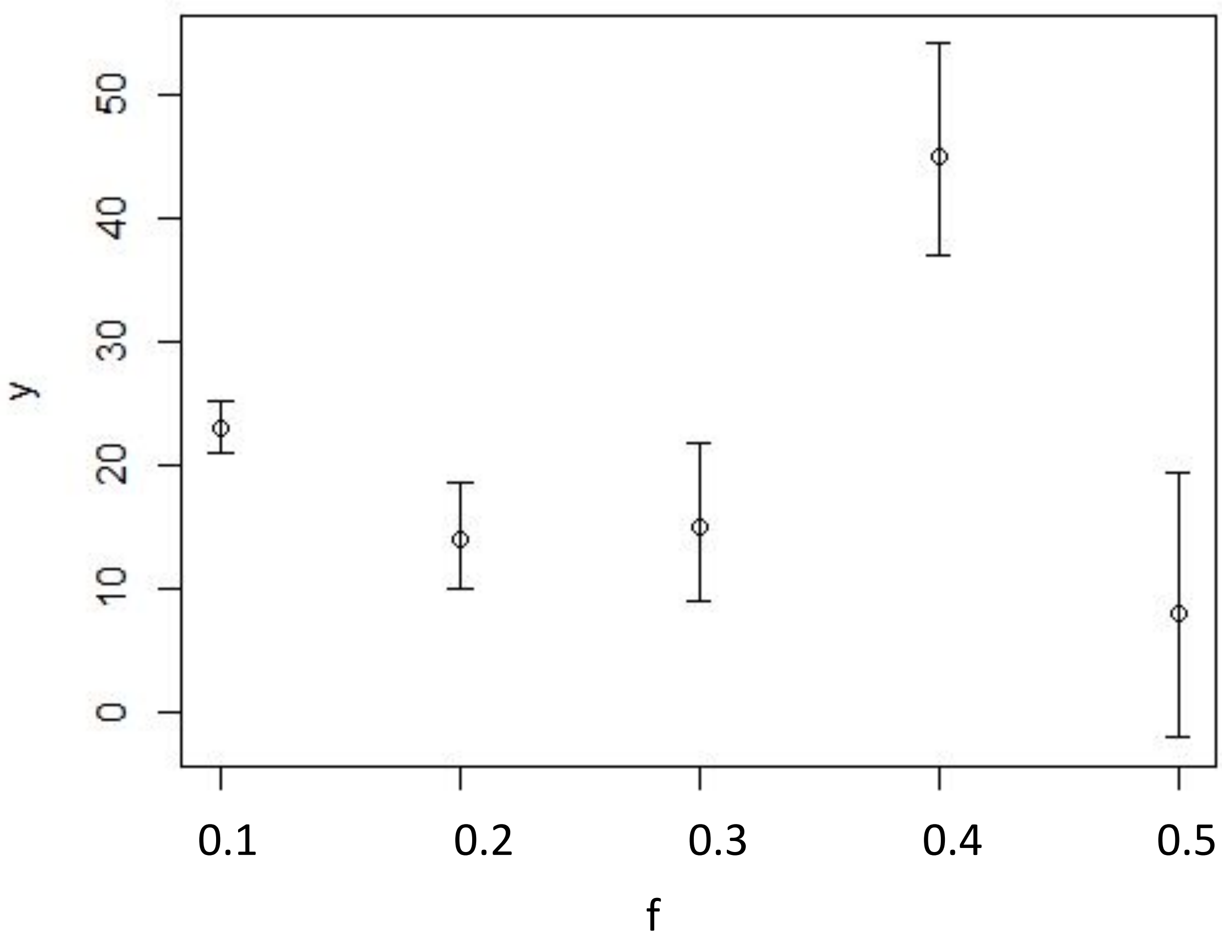
- plot(delta)
- Expect values uniformly distributed around 0
- t.test(delta, conf.level=0.95)
  - [-2.232, +1.14] □ includes zero, so no statistically significant difference between training and testing results



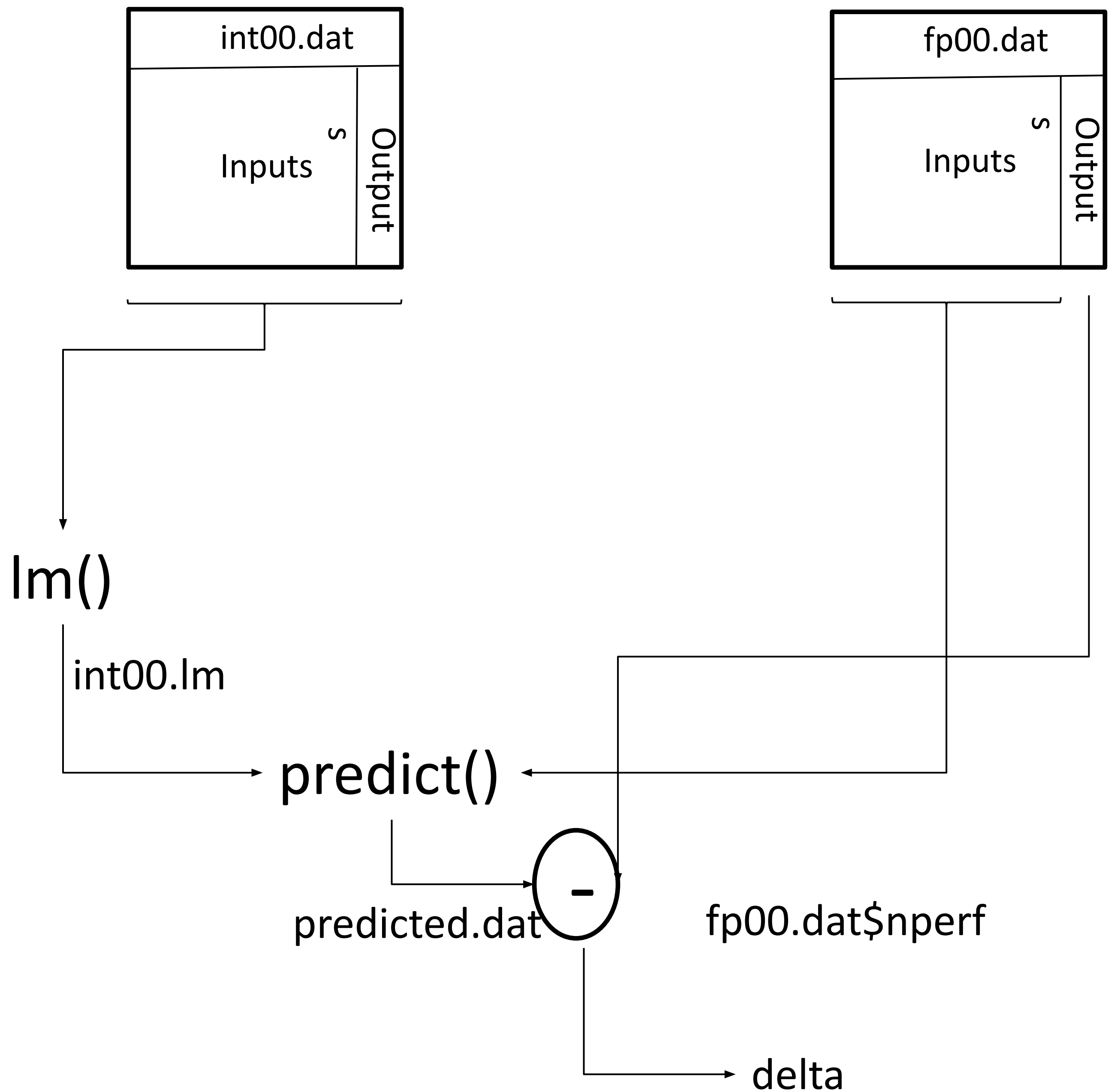
For this lab

- Vector Δ_i = ($predicted_i$ – $actual_i$) for sample i
- New subset of training/testing each time sample() is called
- Repeat k times: $\Delta_1, \Delta_2, \Delta_3, \ldots, \Delta_i, \ldots, \Delta_k$
- Concatenate all Δ_i into one vector: D_f
  - f = train/test partition
- Mean and confidence interval of D_f shows how well model predicted the actual values for k different training-testing samples

Vary *f* to produce this type of plot



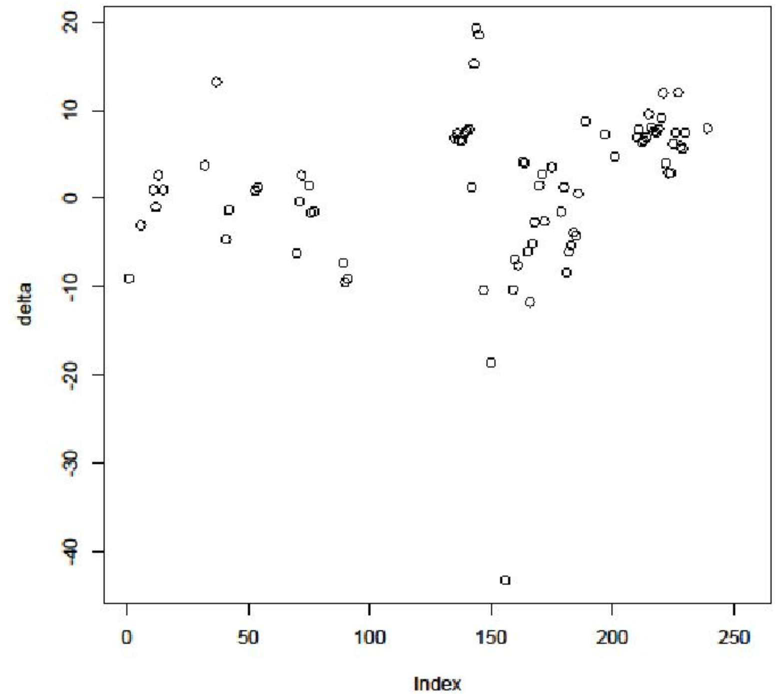
Predicting across data sets



Predicting Fp2000 from Int2000

- plot(delta)
- Expect values uniformly distributed around 0
- t.test(delta, conf.level=0.95)
- [-0.45, +3.40]

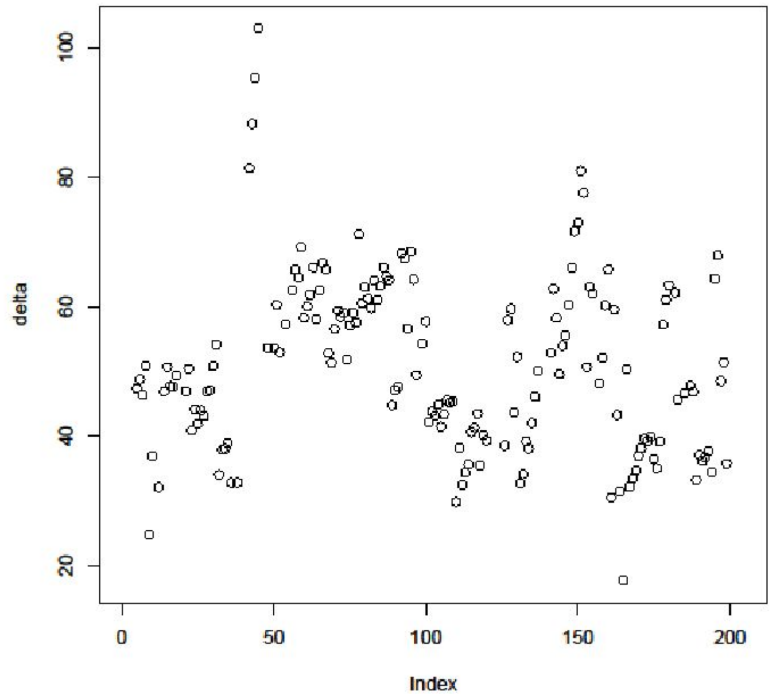
  □ includes zero, so no statistically significant difference between actual and predicted values



Predicting Int2006 from Int2000

- plot(delta)
- Expect values uniformly distributed around 0
- t.test(delta, conf.level=0.95)
- [+48.9, +52.9]

   Far from 0, so significant difference between actual and predicted values



To do

- Read Chapter 5
- Download and complete Lab 5