

# BIOE 498 / BIOE 599: Computational Systems Biology for Medical Applications

## CSE 599V: Advancing Biomedical Models

Lecture 9: Estimating Confidence in Models and  
Parameters: I

Joseph L. Hellerstein\*

Herbert Sauro\*\*

\*eScience Institute, Computer Science & Engineering

\*\*BioEngineering



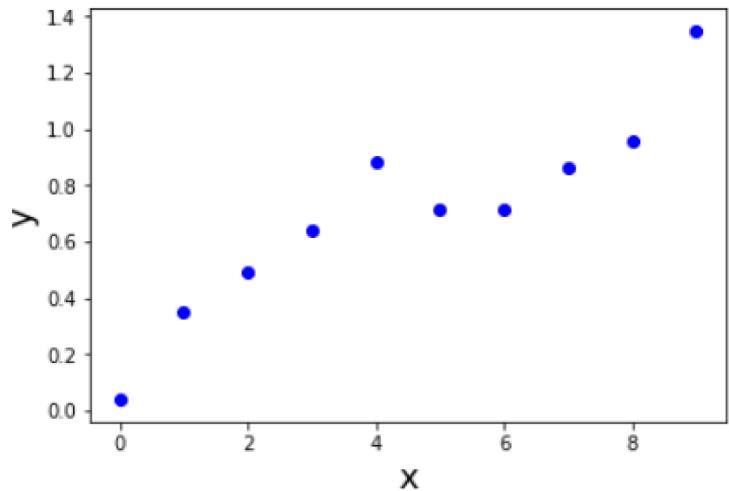
# Agenda

- Residual analysis
- Parameter fitting and evaluation of a mathematical model using least squares regression
- Hypothesis testing to evaluate models

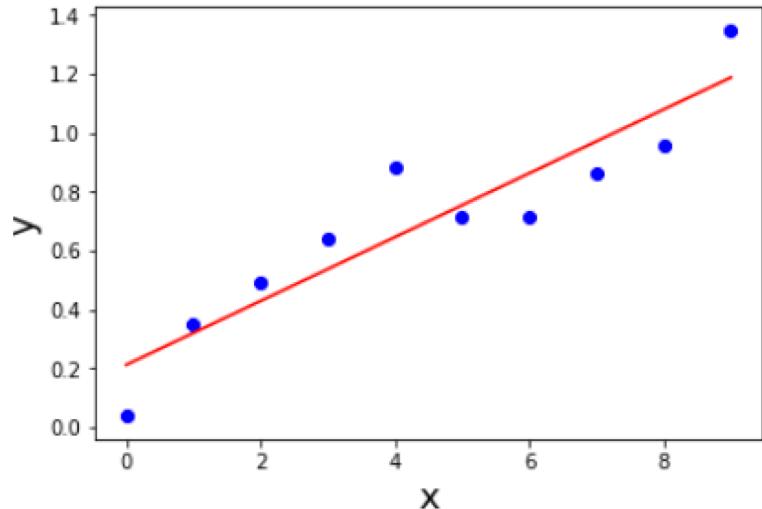


# What is A Good Model?

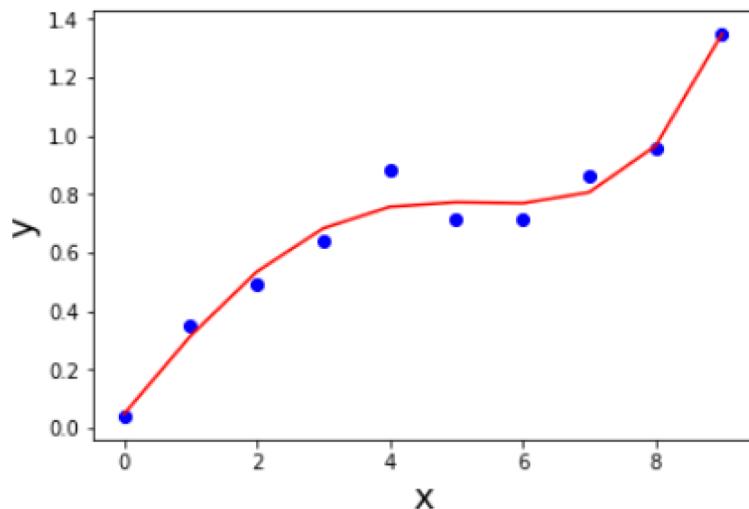
Measured Data



$$y = ax + b$$

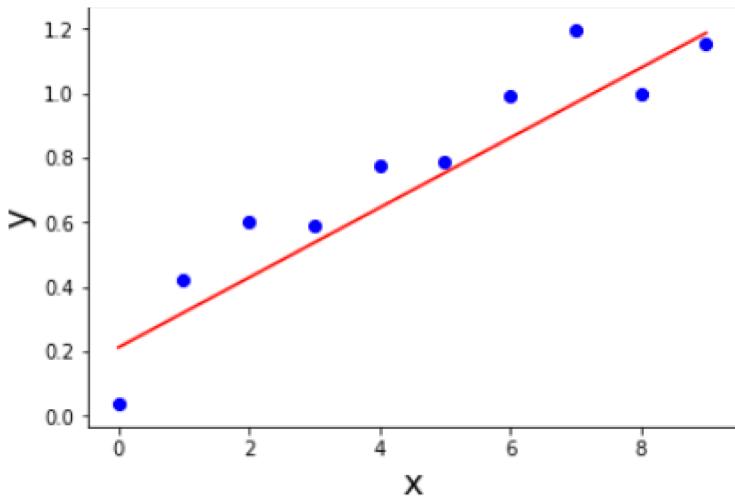
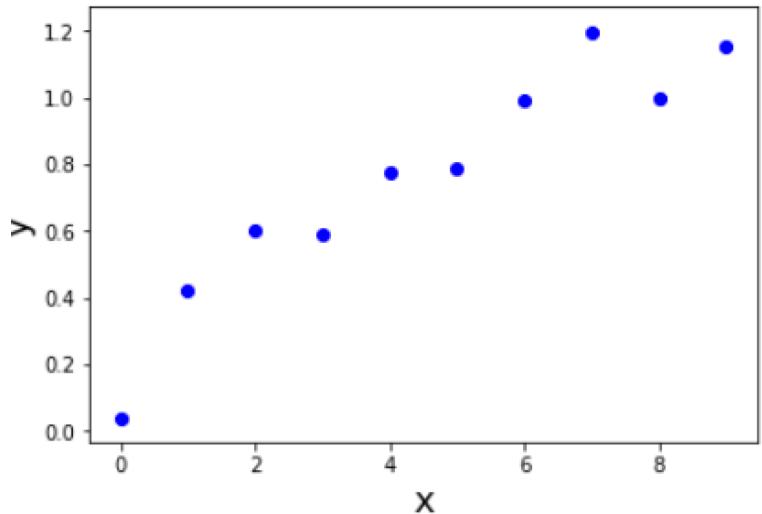


$$y = a_1x + a_2x^2 + a_3x^3 + a_4x^4 + b$$

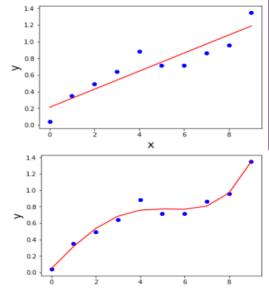
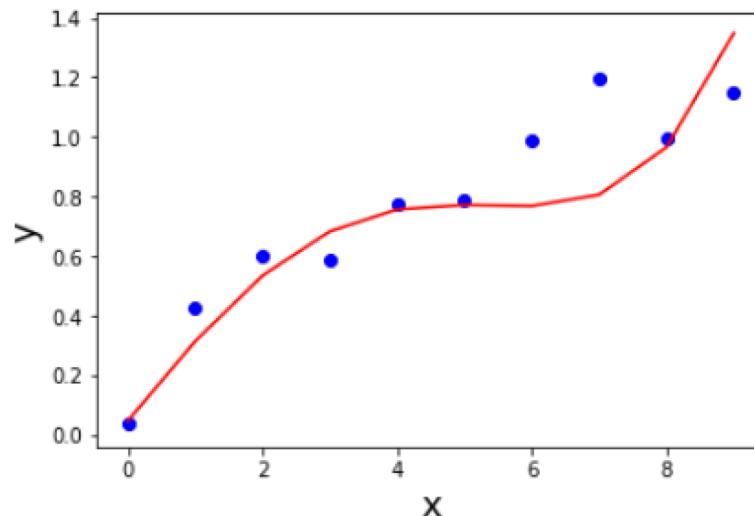


# Generalizing Model to New Data

Data 2



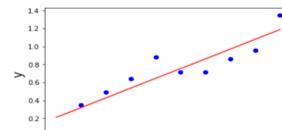
Which is the better model now?



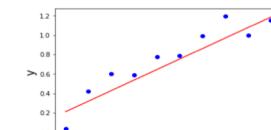
# What is A Good Model?

$$y = ax + b$$

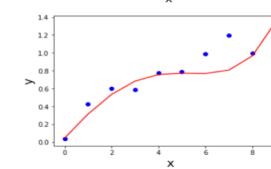
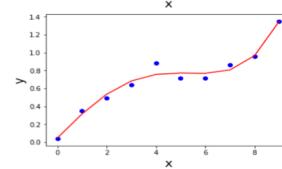
Data 1



Data 2



$$y = a_1x + a_2x^2 + a_3x^3 + a_4x^4 + b$$

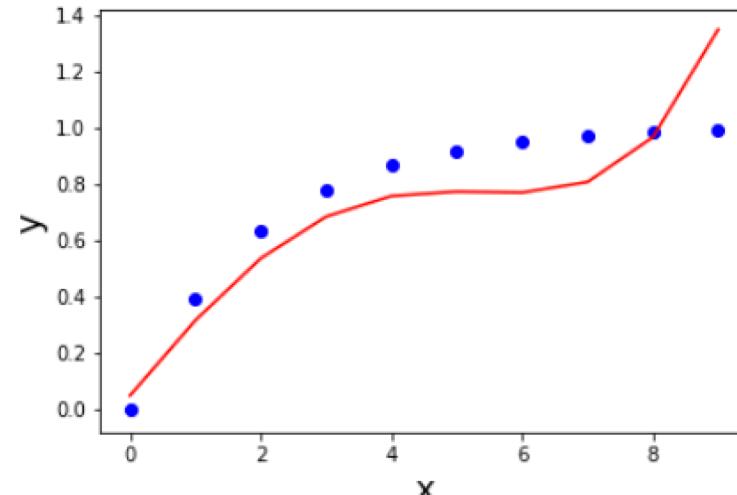
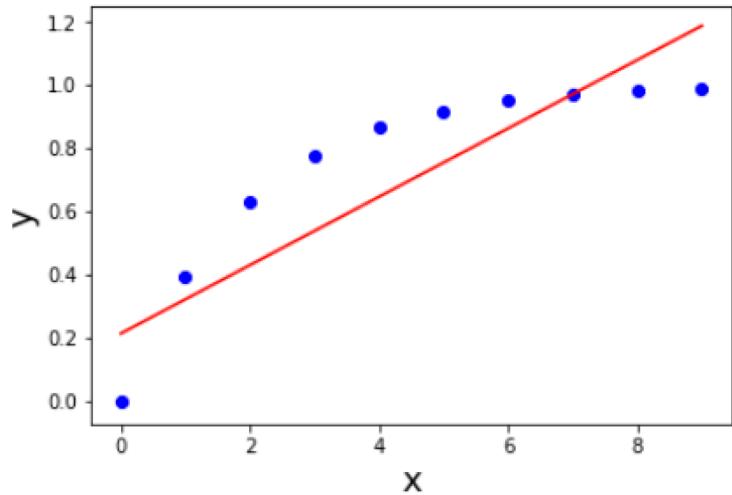
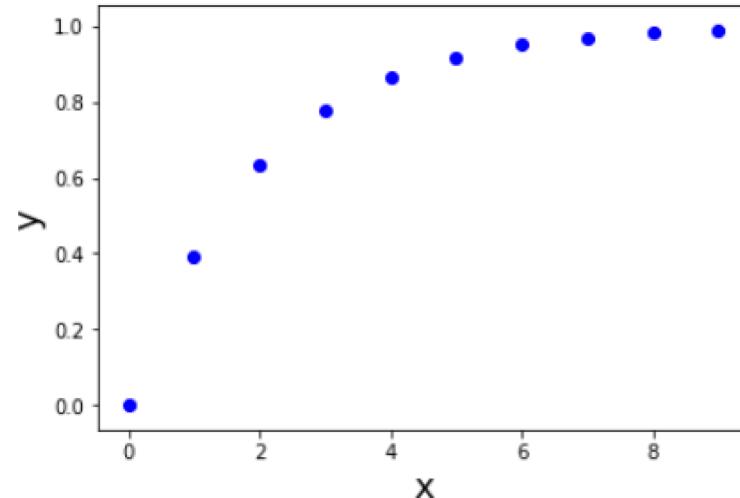


- Fits measurement data
- Simple (e.g., few parameters)
- Generalizes to new measurement data

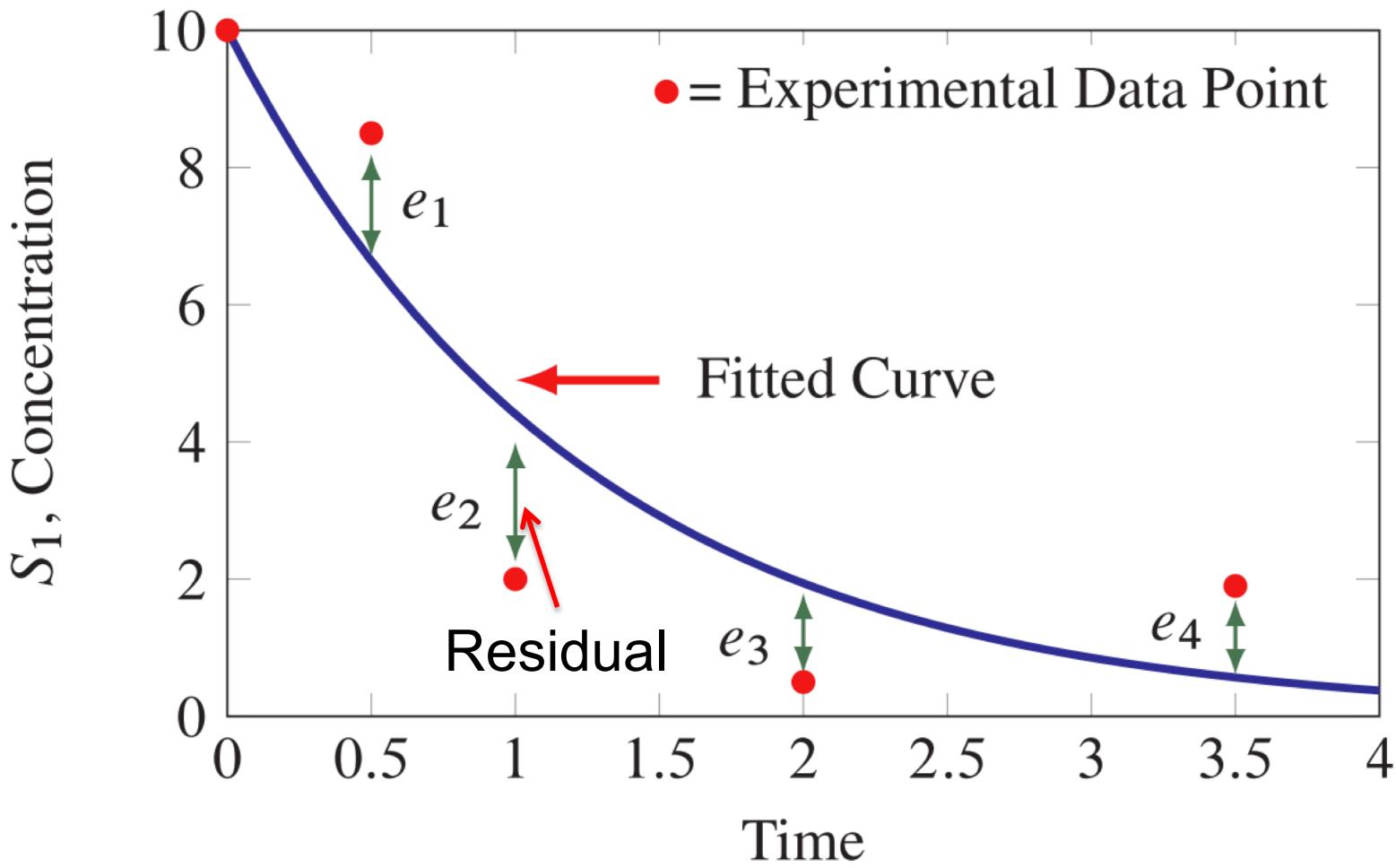


# We Never Know if a Real World Model is Correct!

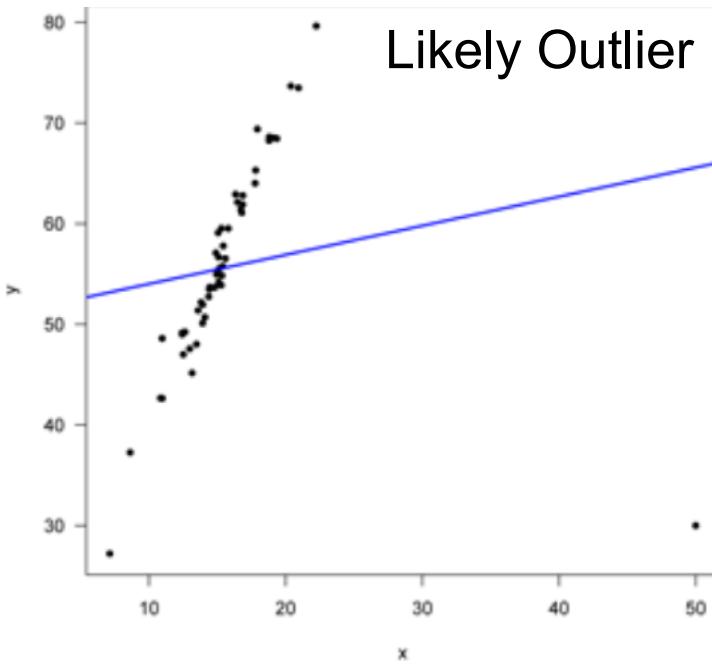
Data Without Measurement Errors



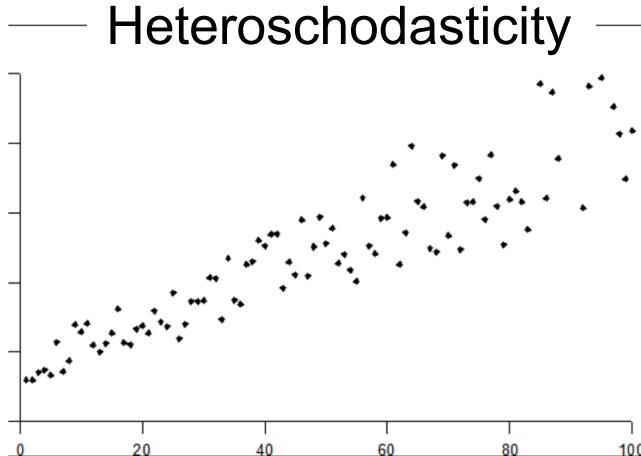
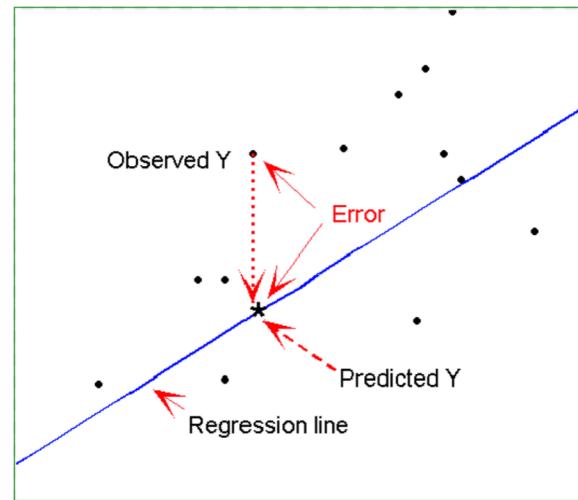
# Residuals = Experimental - Fitted



# What Residuals Tell Us About Models



Functional Bias

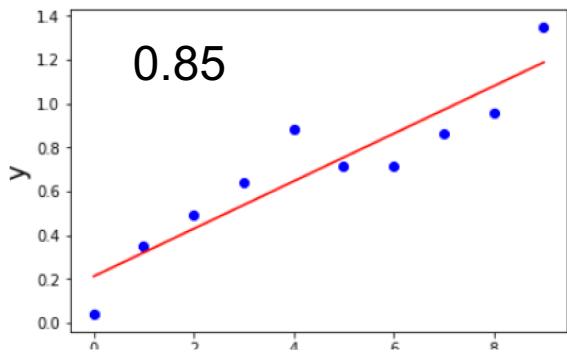


# Quantifying Model Quality: $R^2$

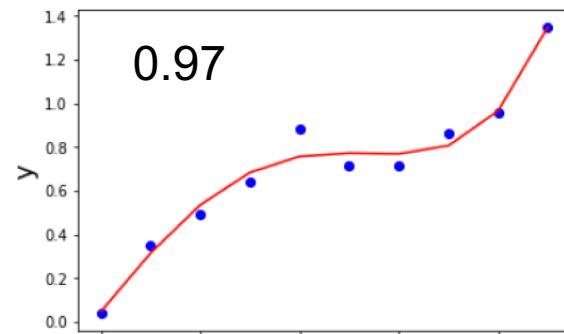
$$R^2 = \frac{var(fitted)}{var(observed)} = 1 - \frac{var(residuals)}{var(observed)}$$

$$y = ax + b$$

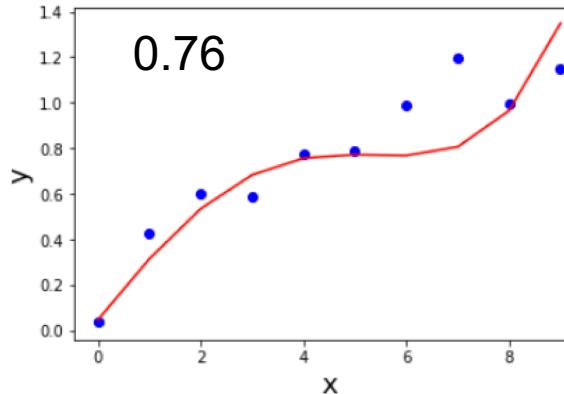
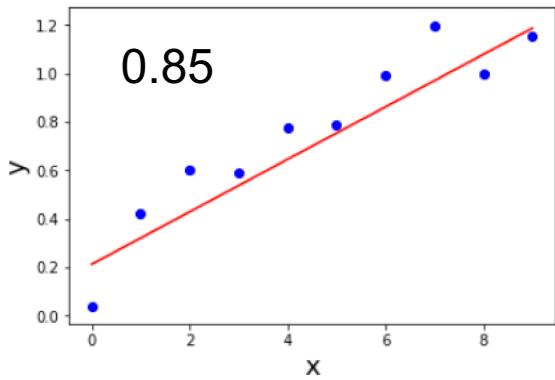
Data 1



$$y = a_1x + a_2x^2 + a_3x^3 + a_4x^4 + b$$



Data 2



# Exercise: Compare Models Using Residuals

## Model 1

$$\begin{aligned} &\rightarrow A; v_0 \\ A &\rightarrow B; k_a A \\ B &\rightarrow C; k_b B \\ C &\rightarrow; k_c C \end{aligned}$$

$$\begin{aligned} v_0 &= 10; k_a = 0.4; \\ k_b &= 0.32; k_c = k_a \end{aligned}$$

## Model 2

$$\begin{aligned} &\rightarrow B; v_0 \\ B &\rightarrow; k_b B \end{aligned}$$

1. Simulate both models
2. Using Model 1 as the “observations” and Model 2 as the “predicted values”, compute the residuals. (Hint: `simulate` returns a matrix of values.)
3. Compute the  $R^2$  of Model 2 with respect to Model 1.
4. Do we achieve the same steady state values?
5. Where are errors the largest? Why?



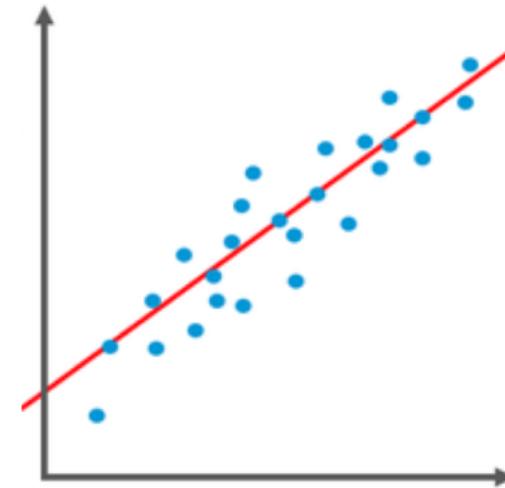
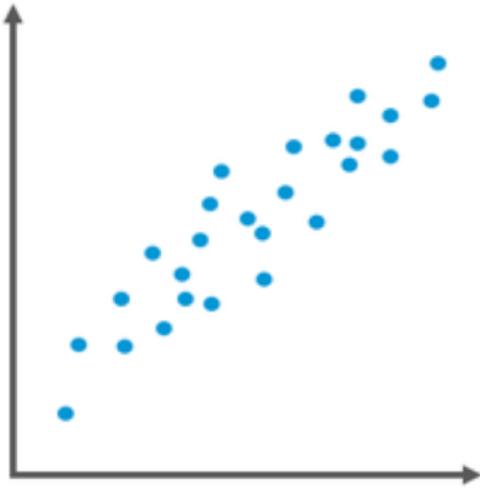
# Linear Least-Squares Regression

## *The Work Horse of Most Modeling Procedures*

- Given dependent variable  $y$ , with values  $y(i)$ , and predictor variables  $x_1, \dots, x_N$  with values  $x_n(i)$ .
- Estimate  $y$  by  $\hat{y} = \sum_n k_n x_n$
- Find the values of  $k_n$  that minimize
$$\sum_i (y(i) - \hat{y}(i))^2$$
- $r(i) = y(i) - \hat{y}(i) = y(i) - k_n x_n(i)$  are the residuals of the regression model.



# How Least Squares Fits a Line to Data

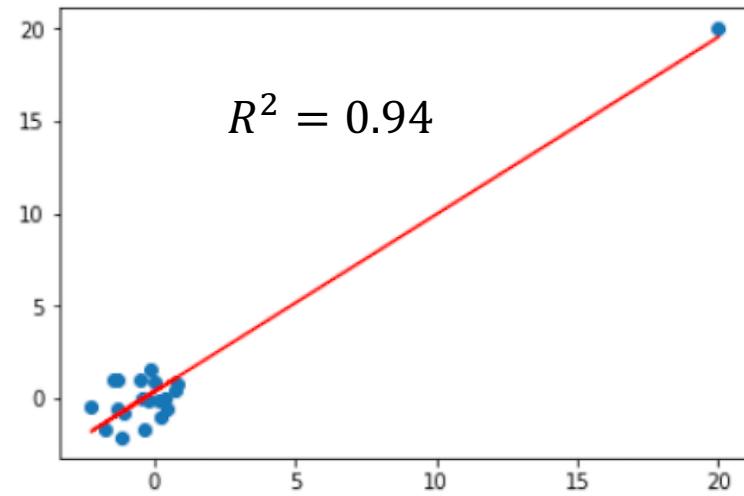
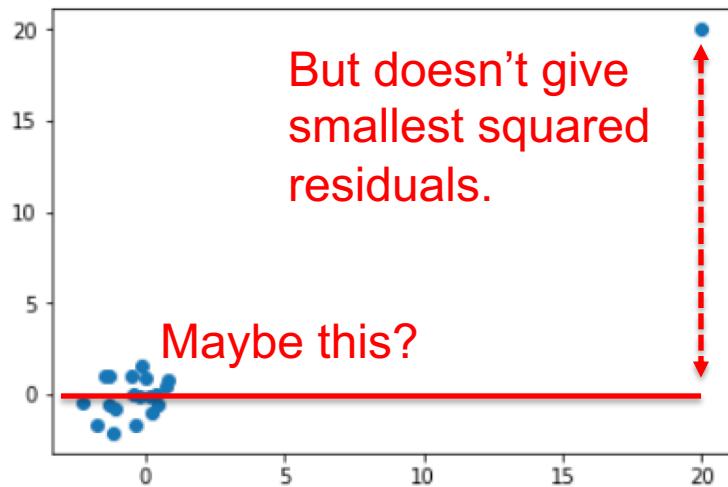


Line minimizes the sum of the squared residuals.



# Limitations of Least Squares

What's the least squares fit  
for these data?

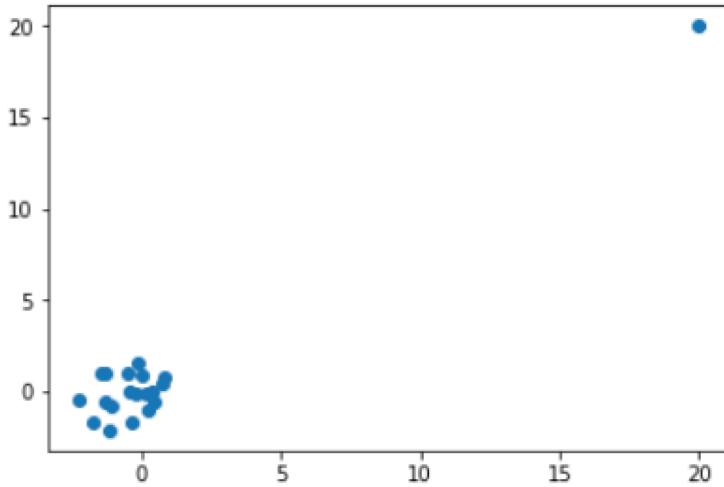


Need outlier detection.



# Linear Regression in Python

## *Prepare the Data*



```
LENGTH = 20
```

```
STD = 1
```

```
# Construct vectors with 19 random values and a 20
```

```
xv = np.random.normal(0, STD, LENGTH - 1)
```

```
xv = np.concatenate([xv, np.array([LENGTH])])
```

```
yv = np.random.normal(0, STD, LENGTH - 1)
```

```
yv = np.concatenate([yv, np.array([LENGTH])])
```

```
# Construct a matrix (has 1's in the first column)
```

```
mat = np.matrix([np.repeat(1, LENGTH), xv])
```

```
mat = mat.transpose() # Transpose to 20 X 2
```

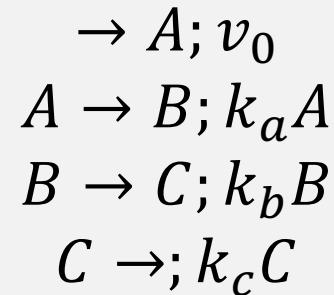
# Linear Regression in Python

## *Fit, Predict, Find Constants, Evaluate*

```
from sklearn import linear_model
from sklearn.metrics import mean_squared_error,
    r2_score
# Fit, find constants evaluate
regr = linear_model.LinearRegression()
regr.fit(mat, yv)
# Predicted values
y_preds = regr.predict(mat)
# R-squared
rsq = r2_score(yv, y_preds)
# Values of the constants
coefs = regr.coef_
#
print("Predictions: ", y_preds)
print("RSQ: ", rsq)
print("Coefficients: ", coefs)
```

# Exercise: Evaluate a Mathematical Model of A System

## Model 1



$$\begin{aligned} v_0 &= 10; k_a = 0.4; \\ k_b &= 0.32; k_c = k_a \end{aligned}$$

## Mathematical Model

$$B = K(1 - e^{-kt})$$

1. Use the simulation of the Model 1 as “observations” by adding a normally distributed error term  $N(0,1)$ .
2. Construct training data and test data using different arrays of normally distributed errors
3. For the mathematical model, estimate  $K, k$  from the training data.



# Hint for Estimating $K, k$

## Mathematical Model

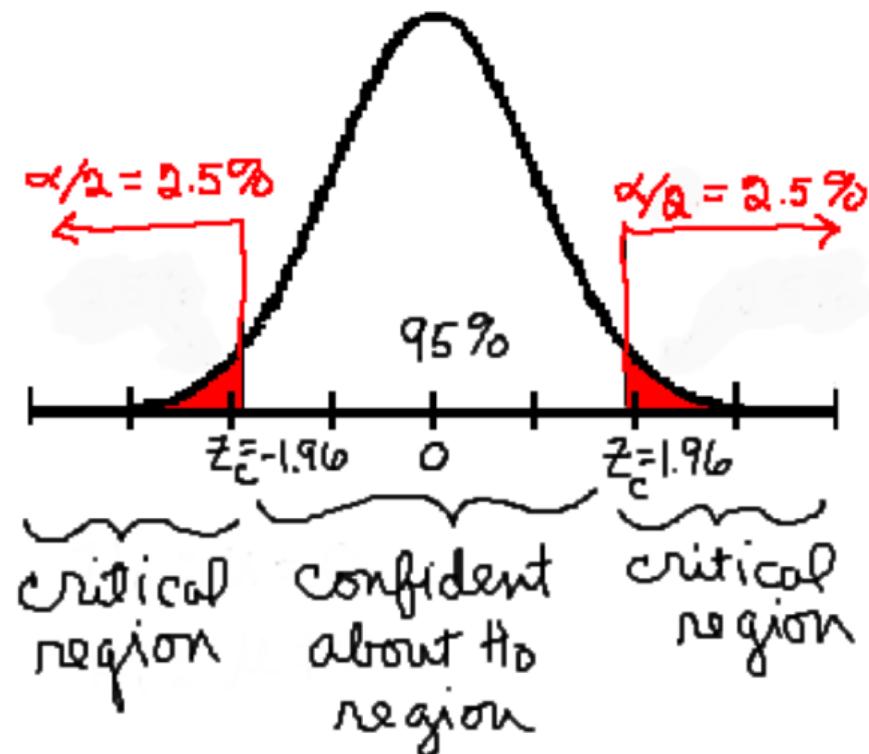
$$B = K(1 - e^{-kt})$$

1. The training data contains values of  $B$ .
2. Note that  $K$  is the steady state value of  $B$ . We can estimate  $K$  from the training data, taking the last  $n$  observations. Call this  $\hat{K}$ .
3. With a little arithmetic,  $\ln(\hat{K} - B) = \ln(\hat{K}) - kt$ .  $k$  can be estimated using linear regression.



# Hypothesis Testing Basics

1. Null Hypothesis: No change from baseline
2. Assume data come from a known distribution (e.g., normal).
3. Calculate a test statistic based on this distribution.
4. If test statistic is near 0 (not in a critical region), do not reject the Null Hypothesis.
5. Otherwise, reject the Null Hypothesis.



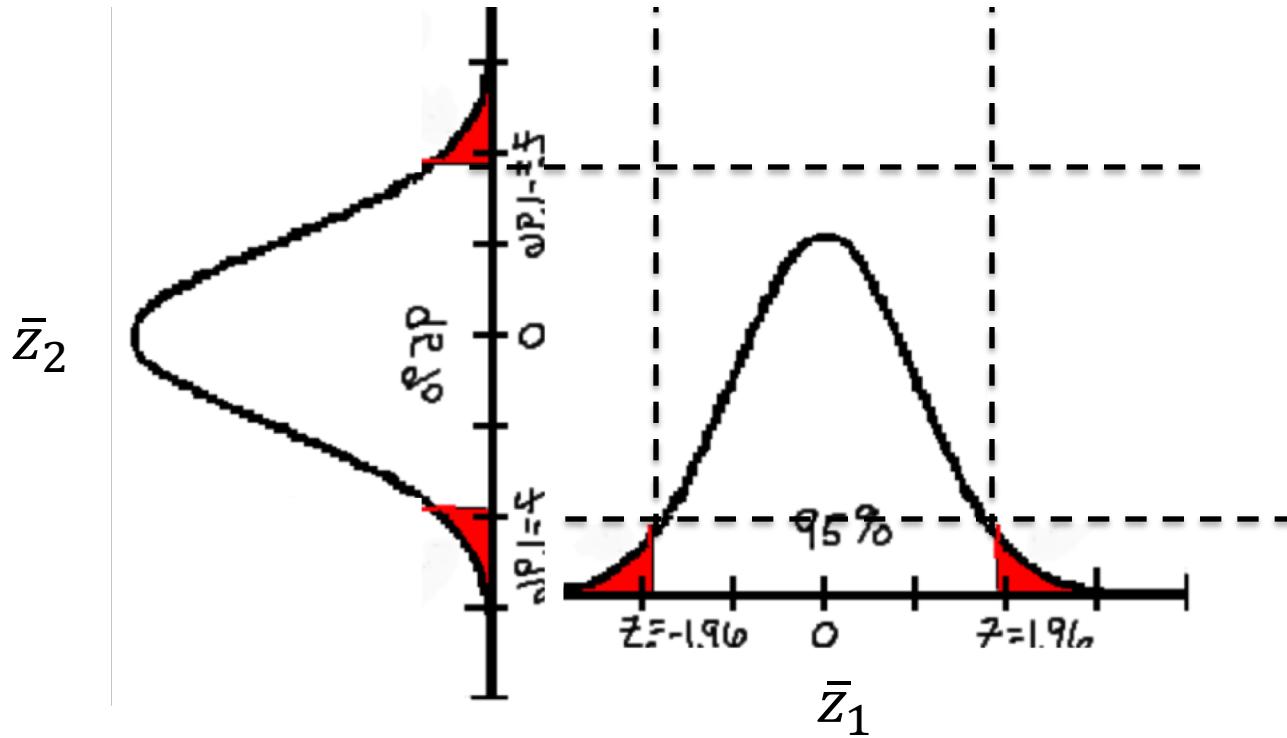
$$H_0: \mu = 0$$

$$H_1: \mu \neq 0$$



# The Multiple Hypothesis Problem

Test the hypothesis that  $\mu_1 = 0$  or  $\mu_2 = 0$



Approaches: Reject the Null Hypothesis if

- ~~1.  $\bar{z}_1$  or  $\bar{z}_2$  is in its critical region.~~
- ~~2.  $\bar{z}_1$  and  $\bar{z}_2$  is in its critical region.~~
3. Use the joint distribution of the test statistics.

