UNIVERSITY of WASHINGTON

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

CSE 599V: Advancing Biomedical Models

Lecture 10: Cross Validation and Bootstrapping

Joseph L. Hellerstein*

Herbert Sauro**

*eScience Institute, Computer Science & Engineering

**BioEngineering

Cross Validation is an efficient way to quantify the quality of a model.

Cross Validation Summary

Divide full data set into *N* Folds

↓

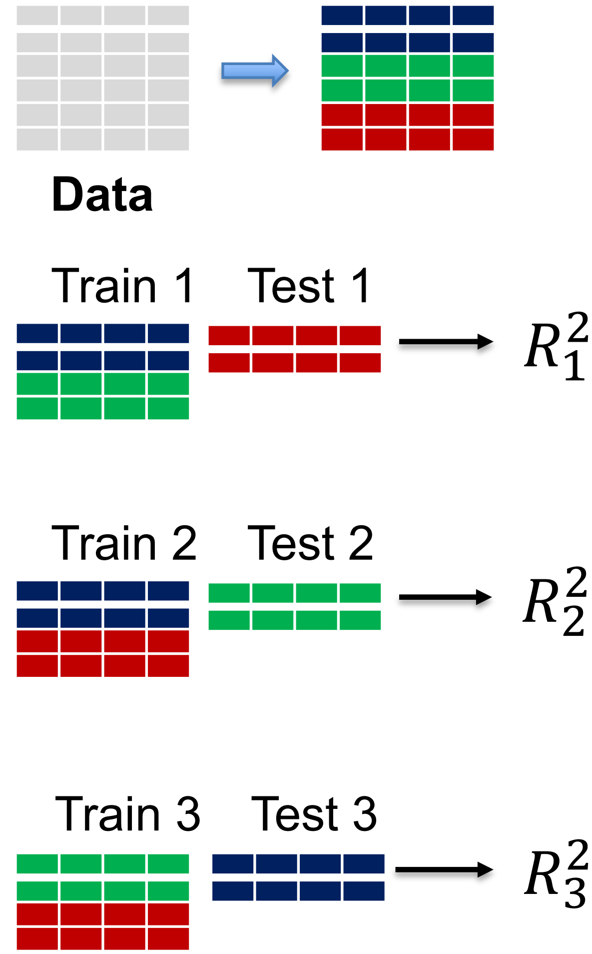
Construct *N* training data sets and *N* test data sets

↓

Obtain *N* evaluations of the model

↓

Report statistics of the evaluations



$$\widehat{R^2} = \frac{R_1^2 + R_2^2 + R_3^2}{3}$$

Python To Generate Indices of Train, Test Data

```
def foldGenerator(num_points, num_folds):
   indices = range(num_points)
   for remainder in range(num_folds):
      test_indices = []
      for idx in indices:
         if idx % num_folds == remainder:
            test_indices.append(idx)
      train_indices = np.array(
         list(set(indices).difference(test_indices)))
      test_indices = np.array(test_indices)
      yield train_indices, test_indices
```

Using the Fold Generator

```
generator = foldGenerator(10, 5)
for g in generator:
    print(g)
```

```
(array([1, 2, 3, 4, 6, 7, 8, 9]), array([0, 5]))
(array([0, 2, 3, 4, 5, 7, 8, 9]), array([1, 6]))
(array([0, 1, 3, 4, 5, 6, 8, 9]), array([2, 7]))
(array([0, 1, 2, 4, 5, 6, 7, 9]), array([3, 8]))
(array([0, 1, 2, 3, 5, 6, 7, 8]), array([4, 9]))
```

Constructing the Regression Matrix

```
def buildMatrix(xv, order):
    """
    :param array-of-float xv:
    :return matrix:
    """
    length = len(xv)
    xv = xv.reshape(length)
    constants = np.repeat(1, length)
    constants = constants.reshape(length)
    data = [constants]
    for n in range(1, order+1):
        data.append(xv*data[-1])
    mat = np.matrix(data)
    return mat.T
```

Doing the Regression

```
def regress(xv, yv, train, test, order=1):
    """
    :param array-of-float xv: predictor values
    :param array-of-float yv: response values
    :param array-of-int train: indices of training data
    :param array-of-int test: indices of test data
    :param int order: Order of the polynomial regression
    return float, array-float, array-float: R2, y_test, y_preds
    """
    regr = linear_model.LinearRegression()
    mat_train = buildMatrix(xv[train], order)
    regr.fit(mat_train, yv[train])
    mat_test = buildMatrix(XV[test], order)
    y_pred = regr.predict(mat_test)
    rsq = r2_score(YV[test], y_pred)
    return rsq, yv[test], y_pred
```

Exercise: Using Cross Validation

Model 1

$$\rightarrow A; v_0$$
$$A \rightarrow B; k_a A$$
$$B \rightarrow C; k_b B$$
$$C \rightarrow; k_c C$$

$$v_0 = 10; k_a = 0.4;$$
$$k_b = 0.32; k_c = k_a$$

Regression Model

$$\hat{B} = b_0 + a_1 t + a_2 t^2 + a_3 t^3$$

1. Use the simulation of the Model 1 as "observations" by adding a normally distributed error term $N(0,1)$.
2. Estimate the quality of the regression model for R^2 using cross validation for 2, 4, and 20 folds.
3. How does the variance of R^2 change with the number of folds?

Calculating the Variance of a Mean Value

- Given i.i.d. random variables $X_1, \cdots X_n$ with variance σ^2, what is the variance of the mean $\bar{X} = \sum_i \frac{X_i}{n}$?
- $Var(\sum_i X_i) = \sum_i Var(X_i) = n\sigma^2$
- $Var\left(\frac{1}{n}X\right) = \frac{1}{n^2}Var(X)$
- So, $Var(\bar{X}) = \frac{\sigma^2}{n}$

Bootstrapping is an efficient way to quantify the uncertainty of parameter estimates.

Bootstrapping Workflow

Fit model to data

↓

Compute Residuals

↓

Generate synthetic response data using the fitted model and randomly selected residuals.

↓

Estimate model parameters for each synthetic data set.

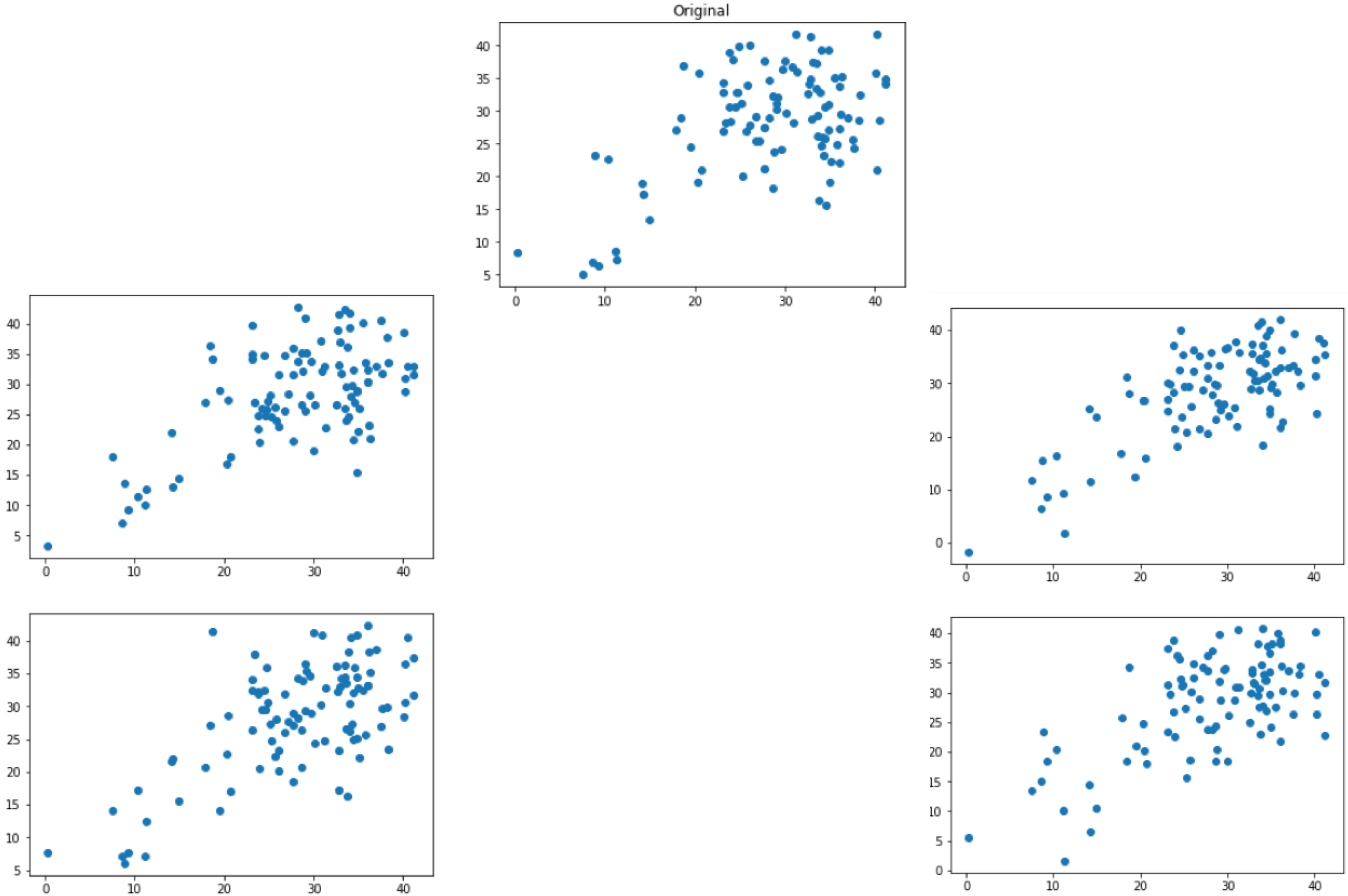
↓

Calculate parameter variances.

Generating a Synthetic Response Data

```
def generateData(y_obs, y_fit):
    """
    :param np.array y_obs
    :param np.array y_fit
    :return np.array: bootstrap data
    """
    residuals = y_obs - y_fit
    length = len(y_obs)
    residuals = residuals.reshape(length)
    samples = np.random.randint(0, length)
    result = y_fit + residuals[samples]
    result = result.reshape(length)
    return result
```

Examples of Synthetic Data



Exercise: Using Bootstrapping

Model 1

$$\rightarrow A; v_0$$
$$A \rightarrow B; k_a A$$
$$B \rightarrow C; k_b B$$
$$C \rightarrow; k_c C$$

$$v_0 = 10; k_a = 0.4;$$
$$k_b = 0.32; k_c = k_a$$

Regression Model

$$\hat{B} = b_0 + a_1 t + a_2 t^2 + a_3 t^3$$

1. Use the simulation of the Model 1 as “observations” by adding a normally distributed error term $N(0,1)$.
2. Use bootstrapping to estimate the variance of parameters
3. How do parameter variances change as you increase the number of synthetic data sets?