Lecture 17: Regularization, Feature Selection, Cross-Validation Tests

GENOME 560, Spring 2017

Su-In Lee, CSE & GS (suinlee@uw.edu)

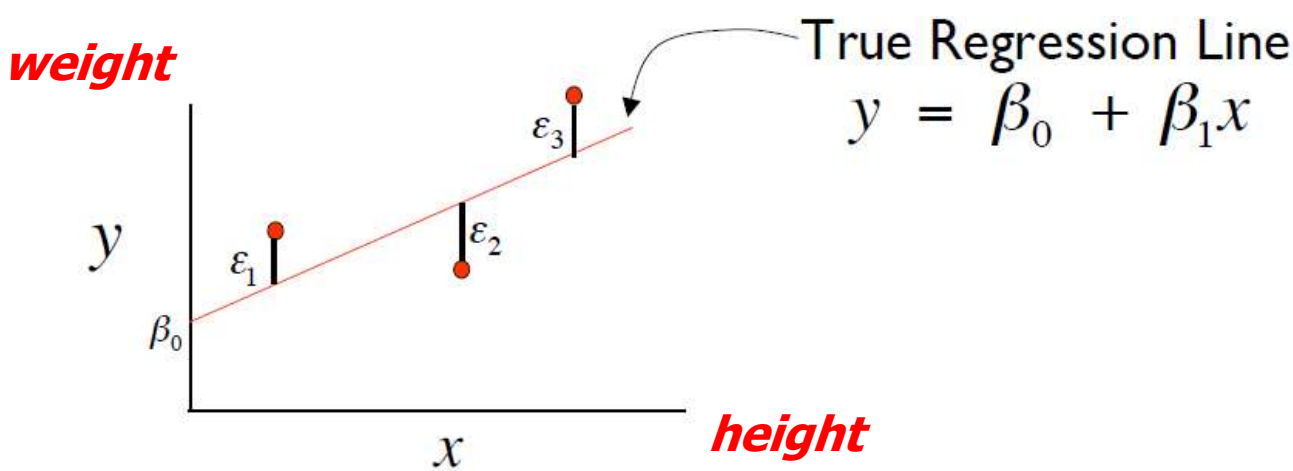
Review of Last Lecture

- Linear regression
  - Linear regression is a probabilistic model.

- **<u>Definition:</u>** There exists parameters β_0, β_1 and σ^2, such that for any fixed value of the predictor variable X, the outcome variable Y is related to X through the model equation:

$$Y = \beta_0 + \beta_1 X + \varepsilon ,$$

where ε is a RV assumed to be $N(0, \sigma^2)$



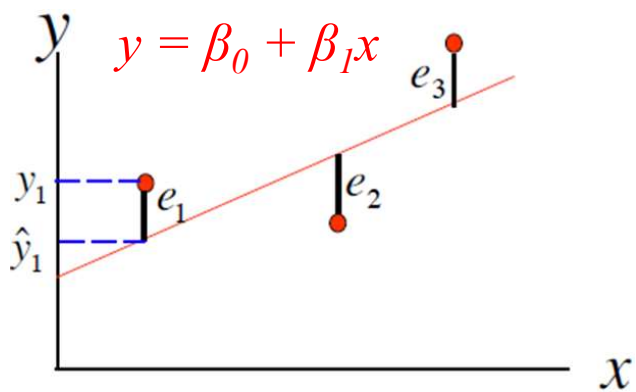
Review of Last Lecture

- Parameter estimation

- ***Least squares***
  - Find β_0 and β_1 that minimizes SSE.

$$f(\beta_0, \beta_1) = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_i)]^2$$

  - Denote the solutions by $\hat{\beta}_0$ and $\hat{\beta}_1$.



- Coefficient of determination (R^2)

Outline

- High-dimensional regression problems
  - Challenges with high-dimensional data
  - Feature selection
- Model selection, Cross validation
  - Cross validation test
  - L1 vs L2 regularization
- R-session
  - Ridge regression using lm.ridge

When there are many predictors

- Say that there are p predictor variables and the input data have n samples ($p > n$)

- Example
  - Outcome variable Y : systolic blood pressure
  - Predictor variables $X_1, \ldots, X_p$: expression levels of p genes
  - Input data: $(y, x_1, x_2, \ldots, x_p)$ from each of n patients

    $(y_1, x_{11}, x_{21}, \ldots, x_{p1})$ from the 1st patient,

    $(y_2, x_{12}, x_{22}, \ldots, x_{p2})$ from the 2nd patient,

    $\vdots$

    $(y_n, x_{1n}, x_{2n}, \ldots, x_{pn})$ from the nth patient

Challenges

- Least squares problems have an infinite number of solutions

- Explanation without using linear algebra
  - There are *more unknowns than equations*
  - Input data: $(y, x_1, x_2, \ldots, x_p)$ from each of *n* patients

$$y_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{21} + \ldots + \beta_p x_{p1} + \varepsilon_1 \longrightarrow \text{1st patient}$$
$$y_2 = \beta_0 + \beta_1 x_{12} + \beta_2 x_{22} + \ldots + \beta_p x_{p2} + \varepsilon_2 \longrightarrow \text{2nd patient}$$
$$\vdots$$
$$y_n = \beta_0 + \beta_1 x_{1n} + \beta_2 x_{2n} + \ldots + \beta_p x_{pn} + \varepsilon_n \longrightarrow n^{\text{th}} \text{ patient}$$

- There are $(p+1)$ unknowns, $\beta_0, \ldots, \beta_p$, and there are *n* equations
  - If $p > n$, There is an infinite number of ways to perfectly fit to the linear model (ε_i=0)

Regularization

- We need more equations or *constraints*

- "Regularized" least squares
  - The cost function is defined as:

$$f(\beta_0, \beta_1, \cdots, \beta_p) = \sum_{i=1}^{n} \left[y_i - (\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}) \right]^2 \boxed{+ \lambda \sum_{j=1}^{p} \beta_j^2}$$

  - *L2 regularization term* encourages to choose $\beta_1, \ldots, \beta_p$ that have small manitude
  - λ : tuning parameter
  - With a large enough λ, there is not an infinite number of solutions when $p > n$

- *L2 regularized linear regression* is also called *Ridge regression*

Feature selection

- Sometimes, we have *many potential predictors* and want to *select relevant predictors* among them

- Each predictor is called a "feature"

- Motivating example
  - Outcome variable Y : systolic blood pressure
  - Predictor variables X_1 , … , X_p : expression levels of *p* genes

  We don't need all genes' expression levels to predict systolic blood pressure.

Feature selection via L1 Regularization

- We want to select a small number of non-zero features

- L1 regularized linear regression
  - The cost function is defined as:

$$f(\beta_0, \beta_1, \cdots, \beta_p) = \sum_{i=1}^{n} \left[y_i - (\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}) \right]^2 \boxed{+ \lambda \sum_{j=1}^{p} \left| \beta_j \right|}$$

  - *L1 regularization term* encourages many of $\beta_1, \ldots, \beta_p$ to be set to zero
  - λ : tuning parameter

- *L1 regularized linear regression* is also called *LASSO regression**

*Rob Tibshirani 1995, Regression shrinkage and selection via the Lasso

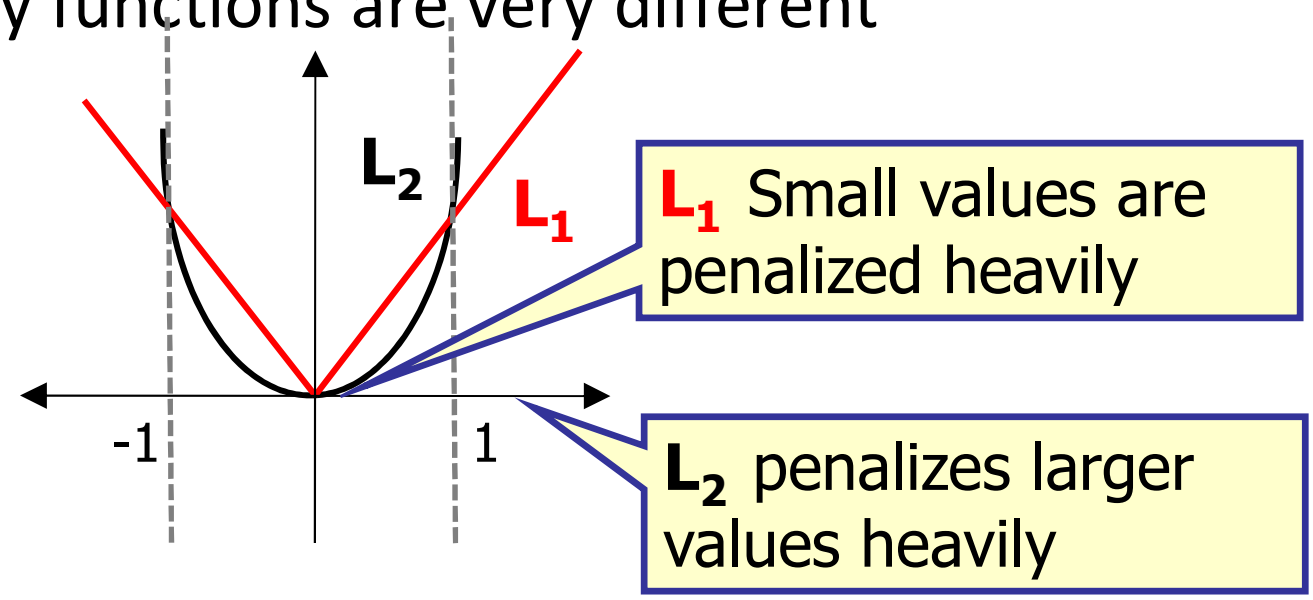
L1 vs. L2 regularization

- Find β's that minimize

L1: $$\sum_{i=1}^{n}\left[y_i-(\beta_0+\beta_1 x_{1i}+\cdots+\beta_p x_{pi})\right]^2+\lambda\sum_{j=1}^{p}\left|\beta_j\right|$$

L2: $$\sum_{i=1}^{n}\left[y_i-(\beta_0+\beta_1 x_{1i}+\cdots+\beta_p x_{pi})\right]^2+\lambda\sum_{j=1}^{p}\beta_j^2$$

- Penalty functions are very different



How to determine λ ?

- L1 regularized linear regression (LASSO)
  - Find β values that minimizes the cost function:

$$f(\beta_0, \beta_1, \cdots, \beta_p) = \sum_{i=1}^{n} \left[y_i - (\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}) \right]^2 \boxed{+ \lambda \sum_{j=1}^{p} \left| \beta_j \right|}$$

  - Large λ: Large penalty term in the objective will force β weights to be set to zero
  - Small λ: Small penalty term will force β weights to be set to zero less strongly than when λ is large

- The tuning parameter λ determines how much β weights can be sparse (having many zeros)

- How do we select the "optimal" tunning parameter λ ?

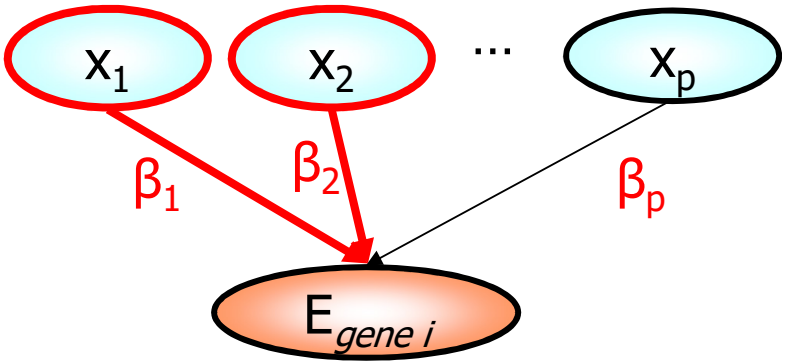
Outline

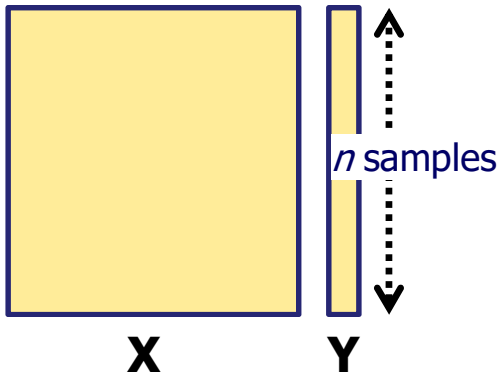
- High-dimensional regression problems
  - Challenges with high-dimensional data
  - Feature selection
- Model selection, Cross validation
  - Cross validation test
  - L1 vs L2 regularization
- R-session
  - Ridge regression using lm.ridge

Cross validation test

- Model selection
  - Which regularization method do we want to use?
  - How do we select the "optimal" tunning parameter λ ?

- Criteria
  - How well each model fits data?
  - How do we estimate the model's "true" error rate?

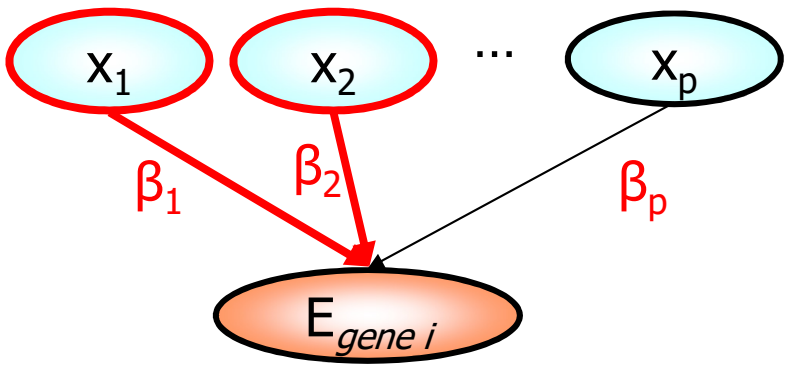


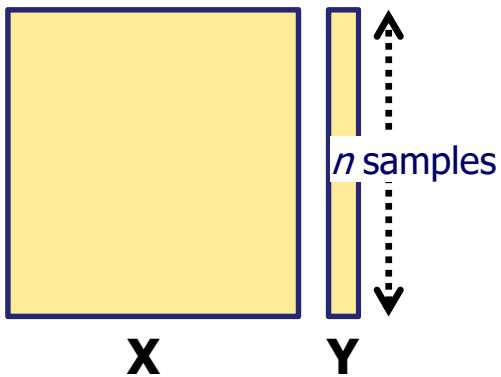


Cross validation test

- If we had access to *an unlimited number of examples*, these questions would have a straightforward answer
  - Choose the model that provides the lowest error rate on the entire population
  - And, of course, that error rate is the true error rate

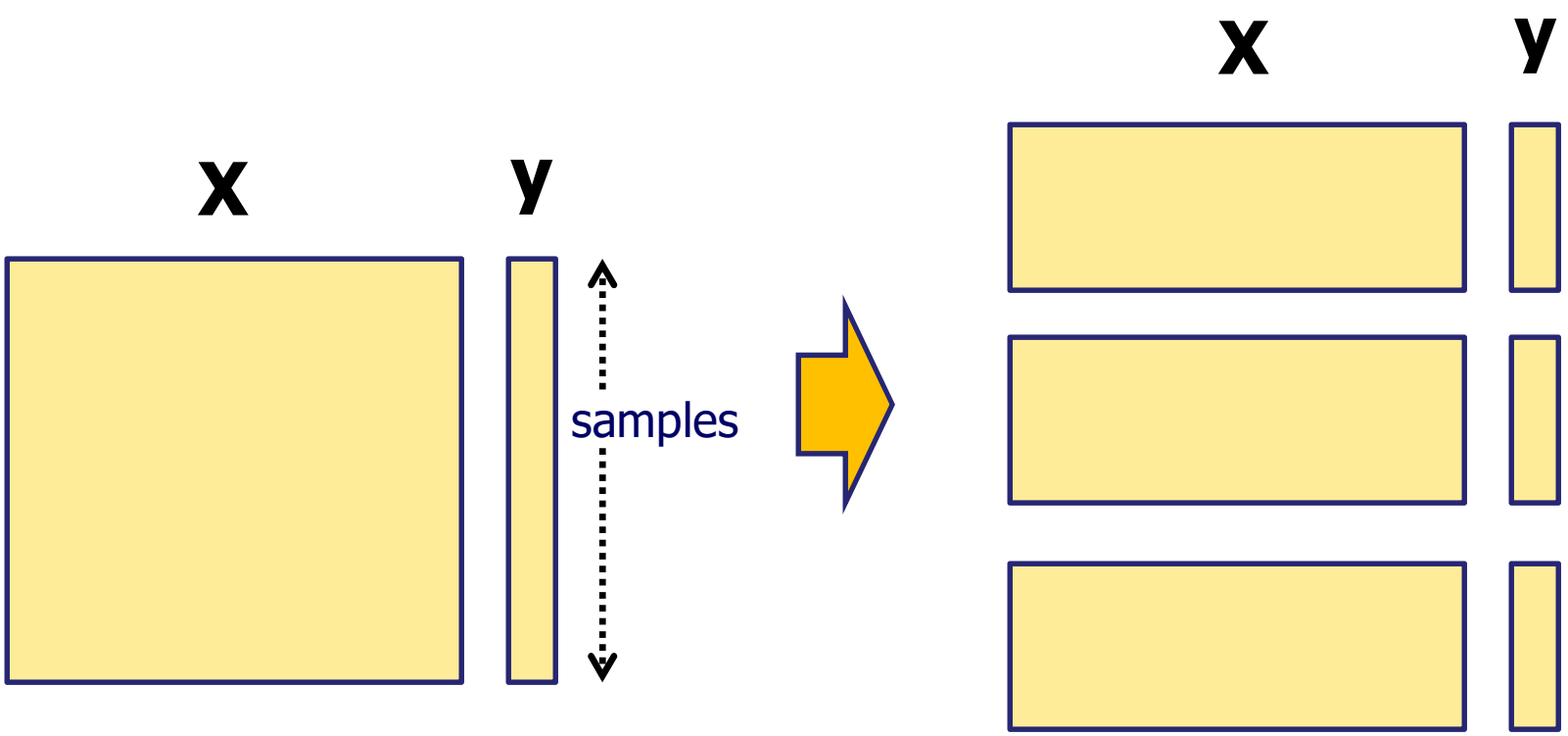
- However, in real applications *only a finite set of examples is available*
  - This number is usually smaller than we would hope for!
  - Why? Data collection is a very expensive process





Cross validation test

- How well the estimated weight values explain the *left out* data (unseen data)?
- Let's divide the samples into *k* (=3) groups



K-fold cross validation test

- Let's divide the samples into *k* (=3) groups

  In each of *k* folds,

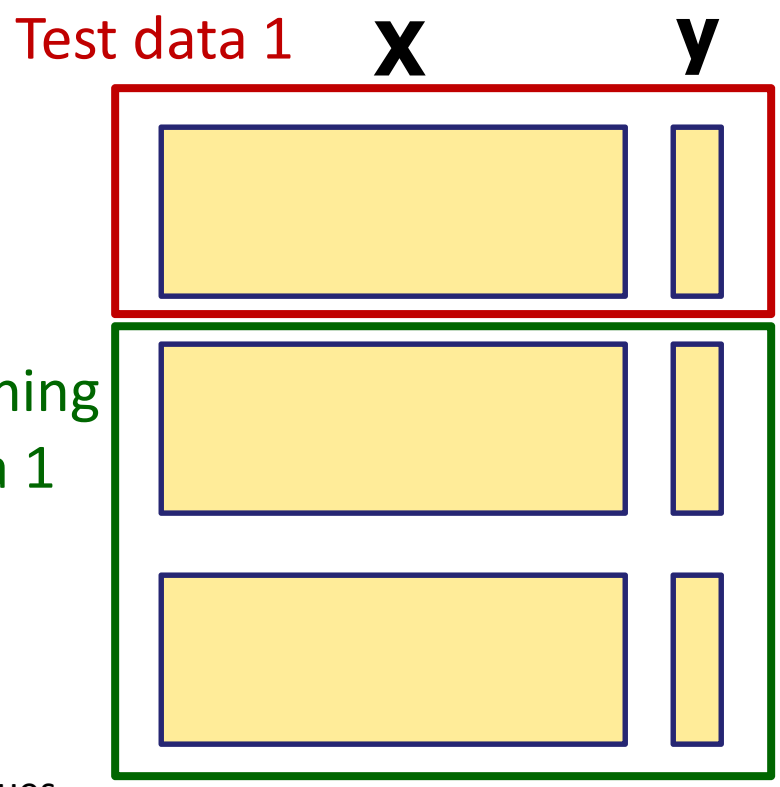
  - **Training:** learning the β values using the Training data 1
  - **Model:** the resulting linear regression model with the learned β values
  - **Test:** compute the error in held-out test data 1
  - We call it "test error"

$$f(\beta_0, \beta_1, \cdots, \beta_p)$$

$$= \sum_{i=1}^{n} \left[y_i - (\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}) \right]^2$$

actual y value

predicted y value based on the learned β values



K-fold cross validation test

- Let's divide the samples into *k* (=3) groups

  In each of *k* folds,

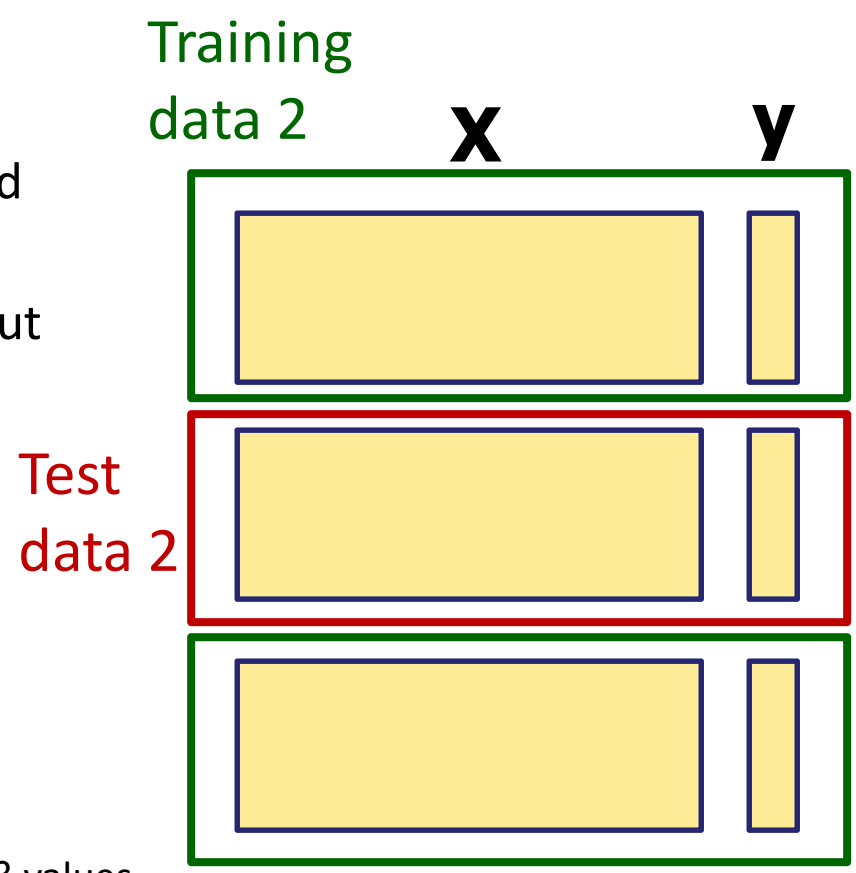
  - **Training:** learning the β values using the Training data 1
  - **Model:** the resulting linear regression model with the learned β values
  - **Test:** compute the error in held-out test data 1
  - We call it "test error"

$$f(\beta_0, \beta_1, \cdots, \beta_p)$$

$$= \sum_{i=1}^{n} \left[y_i - (\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi}) \right]^2$$

actual y value

predicted y value
based on the learned β values



K-fold cross validation test

- Let's divide the samples into *k* (=3) groups

  In each of *k* folds,

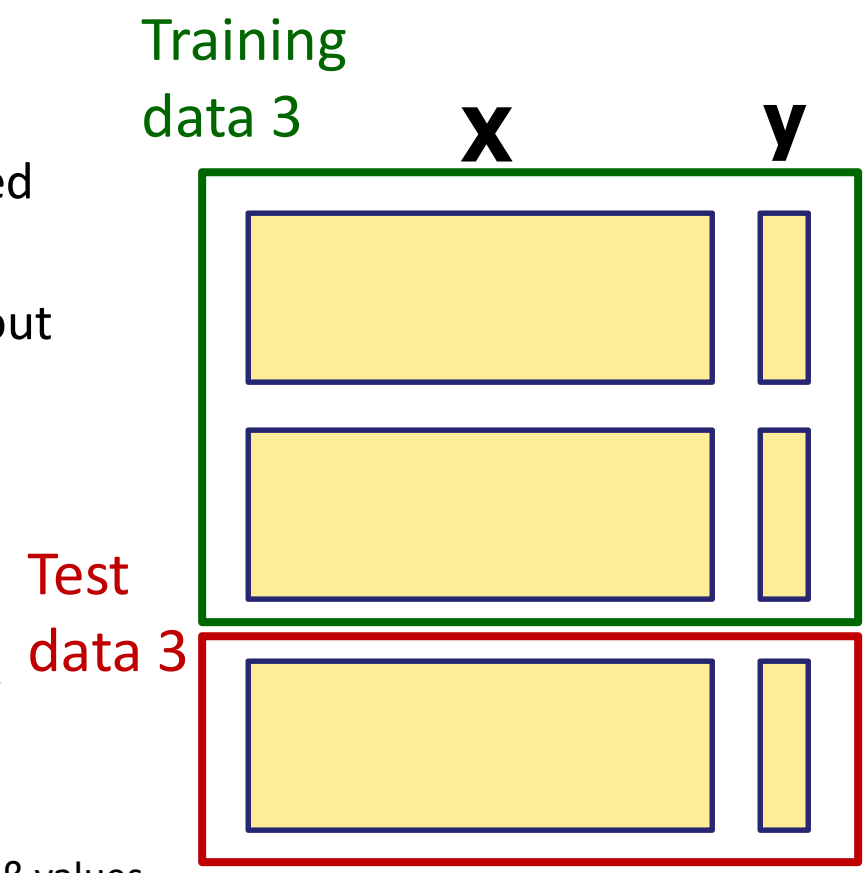
  - **Training:** learning the β values using the Training data 1
  - **Model:** the resulting linear regression model with the learned β values
  - **Test:** compute the error in held-out test data 1
  - We call it "test error"

$$f(\beta_0, \beta_1, \cdots, \beta_p)$$

$$= \sum_{i=1}^{n} \left[y_i - \underline{(\beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi})} \right]^2$$

actual y value

predicted y value
based on the learned β values



k-fold cross-validation test

- In k-fold cross-validation the data is first *partitioned into k equally (or nearly equally) sized segments* or folds.

- Subsequently *k iterations of training and validation* are performed such that within each iteration a different fold of the data is held-out for validation while the remaining k - 1 folds are used for learning.

- The figure in the previous slide demonstrates an example with k = 3.
- In data mining and machine learning 10-fold cross-validation (k = 10) is the most common.

Leave-One-Out Cross Validation

- LOOCV: a special case of k-fold cross-validation where k equals the number of samples in the data.
- In other words in each iteration nearly all the data except for a single observation (sample) are used for training and the model is tested on that single observation.
- An accuracy estimate obtained using LOOCV is known to be almost unbiased.
- It is widely used when the available data are very rare, especially in biology where only dozens of data samples are available.

Cross validation test and model selection

- Model selection
  - Which regularization method do we want to use?
  - How do we select the "optimal" tunning parameter λ ?
- Criteria
  - How well each model fits data?
  - Cross validation is one way to estimate the error
- L1 vs. L2
  - L1 (LASSO): β values have many zeros
  - L2 (Ridge): β values tend to be small but not exactly set to zero
- We can deiced whether to use L1 or L2 based on the prediction error restimation through cross-validation tests
- We can also determine the tunning parameter λ

Outline

- High-dimensional regression problems
  - Challenges with high-dimensional data
  - Feature selection
- Model selection, Cross validation
  - Cross validation test
  - L1 vs L2 regularization
- R-session
  - Ridge regression using lm.ridge

Input Data

- http://homes.cs.washington.edu/~suinlee/genome560/data/mice.txt

- Data on fluctuating proportions of marked cells in marrow from heterozygous Safari cats

- Proportions of cells of one cell type in samples from cats (taken in our department many years ago). Column 1 is the ID number of the particular cat. You will want to plot the data from one cat.
  - For example cat 40004 is rows 1:17, 40005a is 18:31, 40005b is 32:47, 40006 is 48:65, 40665 is 66:83 and so on.

Input Data

- http://homes.cs.washington.edu/~suinlee/genome560/data/mice.txt
  - 1nd column: mouse ID
  - 2rd column: sex
  - 3th column: weight
  - 4th column: length
  - :

mice

F2	sex	weight_g		length_cm		Trigly	Total_Chol	FFA	Insulin_log
F2_1	2	42.8	9.6	14	1646	132	2.974971994		
F2_2	2	38	10.5	14	1646	132	2.974971994		
F2_3	2	33.5	10.8	109	1216	96	2.974971994		
F2_4	1	51.5	11.3	156	1382	147	3.968482949		
F2_5	1	59.2	11.5	261	1563	140	3.968482949		
F2_6	1	51.5	11	134	1823	161	3.968482949		
F2_7	1	56.5	11.2	192	1589	172	4.31255815		
F2_8	1	48.5	10.7	79	1265	85	4.31255815		
F2_9	1	52.7	11.3	96	1375	92	3.646893624		
F2_10	1	58.4	12	134	1567	153	4.347661709		

Our Goals

- We want to learn a Ridge regression model for the insulin level (8th column) using other variables (sex, length, weight, etc.) as predictors.

- We will learn how to diagnose the pairwise correlations between variables.

- We will choose the optimal lambda based on CV errors.