Some Course Admin

For those wanting to do some programming?

- **Assignment 1** By Monday the 2nd of April send me ~ 1 page describing a problem related to your research you would like to tackle with the methods introduced so far in the course.
- In this description include some of the methods/algorithms you would will use and why.
- **Assignment 2** Will obviously be implementing this plan!

Deadline for the homework exercises

- **Deadline for homework sets 1, 2,3** Monday the 2nd of April.
- Note this deadline is only to ensure you get the homework corrected in a timely fashion!

Chapter 3: Linear Methods for Regression

DD3364

March 16, 2012

Introduction: Why focus on these models?

- **Simple and Interpretable**

$$E[Y \mid X] = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p \tag{1}$$

- Can outperform non-linear methods when one has
  - a small number of training examples
  - low signal-to-noise ratio
  - sparse data
- Can be made non-linear by applying a non-linear transformation to the data.

Introduction: Why focus on these models?

- **Simple and Interpretable**

$$E[Y \mid X] = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p \tag{1}$$

- Can outperform non-linear methods when one has
  - a small number of training examples
  - low signal-to-noise ratio
  - sparse data
- Can be made non-linear by applying a non-linear transformation to the data.

Introduction: Why focus on these models?

- **Simple and Interpretable**

$$E[Y \mid X] = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p \tag{1}$$

- Can outperform non-linear methods when one has
  - a small number of training examples
  - low signal-to-noise ratio
  - sparse data
- Can be made non-linear by applying a non-linear transformation to the data.

Linear Regression Models and Least Squares

- Have an input vector $X = (X_1, X_2, \ldots, X_p)^t$.
- Want to predict a real-valued output Y.
- The linear regression has the form

$$f(X) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

How to estimate β:

- **Training data**: $(x_1, y_1), \ldots, (x_n, y_n)$ each $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$
- **Estimate parameters**: Choose β which minimizes

$$\mathrm{RSS}(\beta) = \sum_{i=1}^{n} (y_i - f(x_i))^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2$$

residual sum-of-squares

- Have an input vector $X = (X_1, X_2, \ldots, X_p)^t$.
- Want to predict a real-valued output Y.
- The linear regression has the form

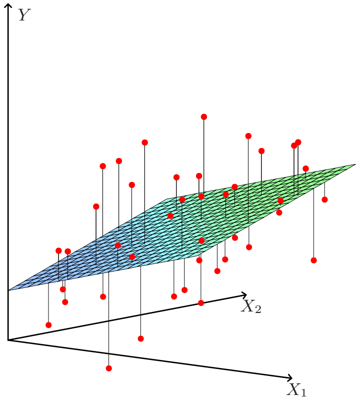
$$f(X) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

How to estimate β:

- **Training data**: $(x_1, y_1), \ldots, (x_n, y_n)$ each $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$
- **Estimate parameters**: Choose β which minimizes

$$\mathrm{RSS}(\beta) = \sum_{i=1}^{n} (y_i - f(x_i))^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2$$

residual sum-of-squares



Find β which minimizes the sum-of-squared residuals from Y.

- **Training data**:
  $(x_1, y_1), \ldots, (x_n, y_n)$ each $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$
- **Estimate parameters**:
  Choose β which minimizes

$$\begin{aligned}\mathrm{RSS}(\beta) &= \sum_{i=1}^{n} (y_i - f(x_i))^2 \\ &= \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2\end{aligned}$$

Minimizing $\mathrm{RSS}(\beta)$

- Re-write

$$\mathrm{RSS}(\beta) = \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2$$

in vector and matrix notation as

$$\mathrm{RSS}(\beta) = (y - \mathbf{X}\beta)^t \, (y - \mathbf{X}\beta)$$

where

$$\beta = (\beta_0, \beta_1, \ldots, \beta_p)^t, \quad \mathbf{X} \quad = \begin{pmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1p} \\ 1 & x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ 1 & x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}$$

and $y = (y_1, \ldots, y_n)^t$.

- Want to find β which minimizes

$$\mathrm{RSS}(\beta) = (y - \mathbf{X}\beta)^t \, (y - \mathbf{X}\beta)$$

- Differentiate $\mathrm{RSS}(\beta)$ w.r.t. β to obtain

$$\frac{\partial\,\mathrm{RSS}}{\partial\beta} = -2\mathbf{X}^t \, (y - \mathbf{X}\,\beta)$$

- Assume $\mathbf{X}$ has **full column rank** $\implies$ is positive definite, set

$$\frac{\partial\,\mathrm{RSS}}{\partial\beta} = -2\mathbf{X}^t \, (y - \mathbf{X}\,\beta) = 0$$

  to obtain the unique solution

$$\hat{\beta} = (X^t X)^{-1} X^t y$$

- Want to find β which minimizes

$$\mathrm{RSS}(\beta) = (y - \mathbf{X}\beta)^t \, (y - \mathbf{X}\beta)$$

- Differentiate $\mathrm{RSS}(\beta)$ w.r.t. β to obtain

$$\frac{\partial \,\mathrm{RSS}}{\partial \beta} = -2\mathbf{X}^t \, (y - \mathbf{X} \, \beta)$$

- Assume $\mathbf{X}$ has **full column rank** $\implies$ is positive definite, set

$$\frac{\partial \,\mathrm{RSS}}{\partial \beta} = -2\mathbf{X}^t \, (y - \mathbf{X} \, \beta) = 0$$

to obtain the unique solution

$$\hat{\beta} = (X^t X)^{-1} X^t y$$

- Want to find β which minimizes

$$\mathrm{RSS}(\beta) = (y - \mathbf{X}\beta)^t \, (y - \mathbf{X}\beta)$$

- Differentiate $\mathrm{RSS}(\beta)$ w.r.t. β to obtain

$$\frac{\partial \, \mathrm{RSS}}{\partial \beta} = -2\mathbf{X}^t \, (y - \mathbf{X} \, \beta)$$

- Assume $\mathbf{X}$ has **full column rank** $\implies$ is positive definite, set

$$\frac{\partial \, \mathrm{RSS}}{\partial \beta} = -2\mathbf{X}^t \, (y - \mathbf{X} \, \beta) = 0$$

  to obtain the unique solution

$$\hat{\beta} = (X^t X)^{-1} X^t y$$

- Given an input x_0 this model predicts its output as

$$\hat{y}_0 = (1, x_0^t)\,\hat{\beta}$$

- The fitted values at the training inputs are

$$\hat{y} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t y$$
$$= H\ y$$

Hat matrix

Estimates using $\hat{\beta}$

- Given an input x_0 this model predicts its output as

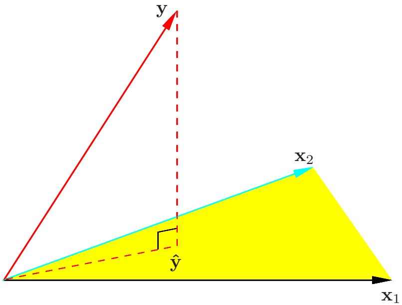
$$\hat{y}_0 = (1, x_0^t)\, \hat{\beta}$$

- The fitted values at the training inputs are

$$\hat{y} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t y$$
$$= \; H \; y$$

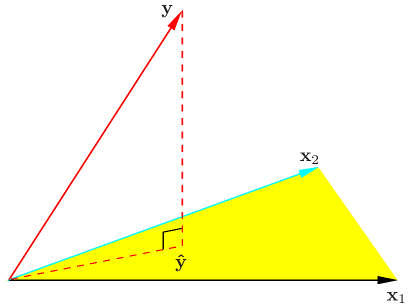
Hat matrix

Geometric interpretation of the least squares estimate



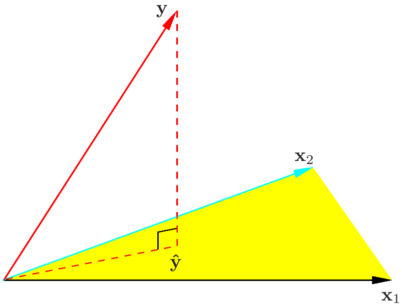
- Let $\mathbf{X}$ be the input data matrix.
- Let $x_{.i}$ be the ith column of $\mathbf{X}$
- In the figure the vector of outputs y is orthogonally projected onto the hyperplane spanned by the vectors $x_{.1}$ and $x_{.2}$.
- The projection $\hat{y}$ represents the least squares estimate.
- The hat matrix H computes the orthogonal projection.

Geometric interpretation of the least squares estimate



- Let $\mathbf{X}$ be the input data matrix.
- Let $x_{.i}$ be the ith column of $\mathbf{X}$
- In the figure the vector of outputs y is orthogonally projected onto the hyperplane spanned by the vectors $x_{.1}$ and $x_{.2}$.
- The projection $\hat{y}$ represents the least squares estimate.
- The hat matrix H computes the orthogonal projection.

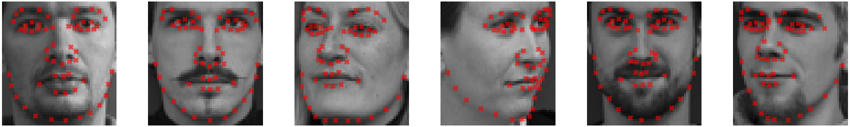
Geometric interpretation of the least squares estimate



- Let $\mathbf{X}$ be the input data matrix.
- Let $x_{.i}$ be the ith column of $\mathbf{X}$
- In the figure the vector of outputs y is orthogonally projected onto the hyperplane spanned by the vectors $x_{.1}$ and $x_{.2}$.
- The projection $\hat{y}$ represents the least squares estimate.
- The hat matrix H computes the orthogonal projection.

An example

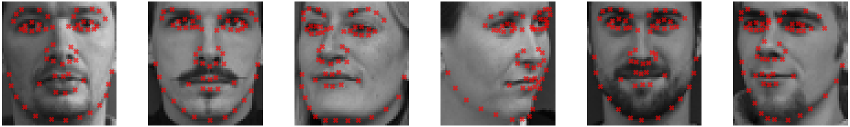
Regression Example: Face Landmark Estimation



Example of training data

- Have training data in the following format.
  - **Input:** image of fixed size of a face ($W \times H$ matrix of pixel intensities $=$ vector of length WH)
  - **Output:** coordinates of F facial features of the face
- Want to learn F linear regression functions f_i
- f_i maps the image vector to x-coord of the ith facial feature.
- Learn also F regression fns g_i for the y-coord.

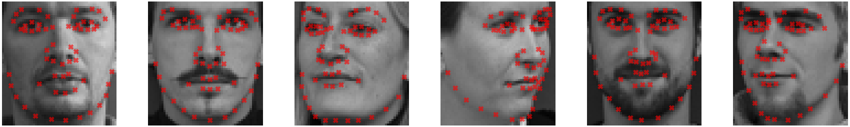
Regression Example: Face Landmark Estimation



Example of training data

- Have training data in the following format.
  - **Input:** image of fixed size of a face ($W \times H$ matrix of pixel intensities $=$ vector of length WH)
  - **Output:** coordinates of F facial features of the face
- Want to learn F linear regression functions f_i
- f_i maps the image vector to x-coord of the ith facial feature.
- Learn also F regression fns g_i for the y-coord.

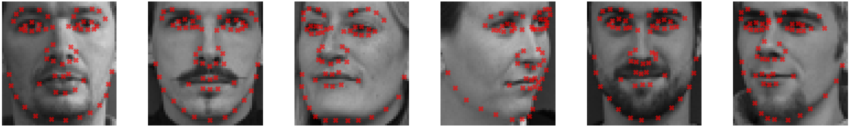
Regression Example: Face Landmark Estimation



Example of training data

- Have training data in the following format.
  - **Input:** image of fixed size of a face ($W \times H$ matrix of pixel intensities $=$ vector of length WH)
  - **Output:** coordinates of F facial features of the face
- Want to learn F linear regression functions f_i
- f_i maps the image vector to x-coord of the ith facial feature.
- Learn also F regression fns g_i for the y-coord.

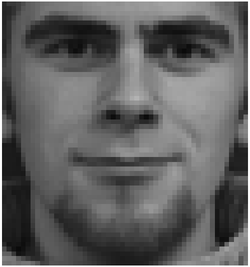
Regression Example: Face Landmark Estimation



Example of training data

- Have training data in the following format.
  - **Input:** image of fixed size of a face ($W \times H$ matrix of pixel intensities = vector of length WH)
  - **Output:** coordinates of F facial features of the face
- Want to learn F linear regression functions f_i
- f_i maps the image vector to x-coord of the ith facial feature.
- Learn also F regression fns g_i for the y-coord.

Regression Example: Face Landmark Estimation

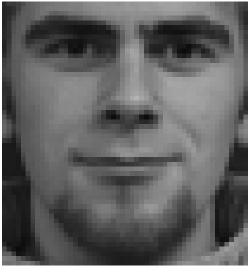


$\xrightarrow{f_{14},\ g_{14}}$

Input **Output**

- Given a test image want to predict each of its facial landmark points.
- How well can ordinary least squares regression do on this problem?

Regression Example: Face Landmark Estimation



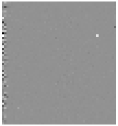
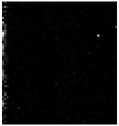
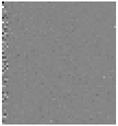
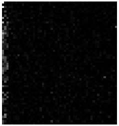
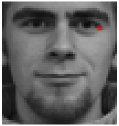
Input

$$\xrightarrow{f_{14},\ g_{14}}$$

Output

- Given a test image want to predict each of its facial landmark points.
- How well can ordinary least squares regression do on this problem?

Landmark estimation using ols regression

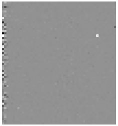
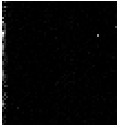
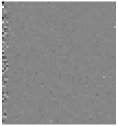
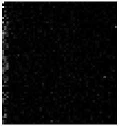
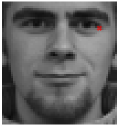
$\hat{\beta}_{x,14}$	$\lvert\hat{\beta}_{x,14}\rvert$	$\hat{\beta}_{y,14}$	$\lvert\hat{\beta}_{y,14}\rvert$	Estimated Landmark on novel image
				

These are not promising weight vectors!

Estimate not even in image

- This problem is too hard for ols regression and it fails miserably.
- p is too large and many of the x_i are highly correlated.

Landmark estimation using ols regression

$\hat{\beta}_{x,14}$	$\lvert\hat{\beta}_{x,14}\rvert$	$\hat{\beta}_{y,14}$	$\lvert\hat{\beta}_{y,14}\rvert$	Estimated Landmark on novel image
				

These are not promising weight vectors!

Estimate not even in image

- This problem is too hard for ols regression and it fails miserably.
- p is too large and many of the x_i are highly correlated.

Singular $\mathbf{X}^t\mathbf{X}$

When $\mathbf{X}$ is not full rank?

- Not all the columns of $\mathbf{X}$ are linearly independent.
- In this case $\mathbf{X}^t\mathbf{X}$ is singular $\implies$ $\hat{\beta}$ not uniquely defined.
- The fitted values $\hat{y} = \mathbf{X}\hat{\beta}$ are still the projection of y onto the column space of $\mathbf{X}$ but $\exists \gamma \neq \hat{\beta}$ such that

$$\hat{y} = \mathbf{X}\hat{\beta} = \mathbf{X}\gamma$$

- **Non-full-rank** case occurs when
  - one or more of the qualitative inputs are encoded redundantly,
  - when the number of inputs $p > n$ the number of training examples.

- Not all the columns of $\mathbf{X}$ are linearly independent.
- In this case $\mathbf{X}^t\mathbf{X}$ is singular $\implies$ $\hat{\beta}$ not uniquely defined.
- The fitted values $\hat{y} = \mathbf{X}\hat{\beta}$ are still the projection of y onto the column space of $\mathbf{X}$ but $\exists \gamma \neq \hat{\beta}$ such that

$$\hat{y} = \mathbf{X}\hat{\beta} = \mathbf{X}\gamma$$

- **Non-full-rank** case occurs when
  - one or more of the qualitative inputs are encoded redundantly,
  - when the number of inputs $p > n$ the number of training examples.

When $\mathbf{X}$ is not full rank?

- Not all the columns of $\mathbf{X}$ are linearly independent.
- In this case $\mathbf{X}^t\mathbf{X}$ is singular $\implies$ $\hat{\beta}$ not uniquely defined.
- The fitted values $\hat{y} = \mathbf{X}\hat{\beta}$ are still the projection of y onto the column space of $\mathbf{X}$ but $\exists\, \gamma \neq \hat{\beta}$ such that

$$\hat{y} = \mathbf{X}\hat{\beta} = \mathbf{X}\gamma$$

- **Non-full-rank** case occurs when
  - one or more of the qualitative inputs are encoded redundantly,
  - when the number of inputs $p > n$ the number of training examples.

When $\mathbf{X}$ is not full rank?

- Not all the columns of $\mathbf{X}$ are linearly independent.
- In this case $\mathbf{X}^t\mathbf{X}$ is singular $\implies$ $\hat{\beta}$ not uniquely defined.
- The fitted values $\hat{y} = \mathbf{X}\hat{\beta}$ are still the projection of y onto the column space of $\mathbf{X}$ but $\exists\, \gamma \neq \hat{\beta}$ such that

$$\hat{y} = \mathbf{X}\hat{\beta} = \mathbf{X}\gamma$$

- **Non-full-rank** case occurs when
  - one or more of the qualitative inputs are encoded redundantly,
  - when the number of inputs $p > n$ the number of training examples.

What can we say about the distribution of $\hat{\beta}$?

Analysis of the distribution of $\hat{\beta}$.

- This requires making some assumptions. These are
  - the observations y_i are uncorrelated
  - y_i have constant variance σ^2 **and**
  - x_i are fixed (non-random) $\leftarrow$ this make analysis easier
- The covariance matrix of $\hat{\beta}$ is then

$$\begin{aligned}\mathrm{Var}(\hat{\beta}) = \mathrm{Var}((\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t y) &= (\mathbf{X}^t\mathbf{X})^{-1} X^t \,\mathrm{Var}(y)\, X\,(\mathbf{X}^t\mathbf{X})^{-1} \\ &= (\mathbf{X}^t\mathbf{X})^{-1}\sigma^2\end{aligned}$$

- Usually one estimates the variance σ^2 with

$$\hat{\sigma}^2 = \frac{1}{n-p-1}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Analysis of the distribution of $\hat{\beta}$.

- This requires making some assumptions. These are
  - the observations y_i are uncorrelated
  - y_i have constant variance σ^2 **and**
  - x_i are fixed (non-random) $\leftarrow$ this make analysis easier
- The covariance matrix of $\hat{\beta}$ is then

$$\begin{aligned}\mathrm{Var}(\hat{\beta}) = \mathrm{Var}((\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t y) &= (\mathbf{X}^t\mathbf{X})^{-1} X^t \,\mathrm{Var}(y)\, X (\mathbf{X}^t\mathbf{X})^{-1} \\ &= (\mathbf{X}^t\mathbf{X})^{-1}\sigma^2\end{aligned}$$

- Usually one estimates the variance σ^2 with

$$\hat{\sigma}^2 = \frac{1}{n-p-1}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Analysis of the distribution of $\hat{\beta}$.

- This requires making some assumptions. These are
  - the observations y_i are uncorrelated
  - y_i have constant variance σ^2 **and**
  - x_i are fixed (non-random) $\leftarrow$ this make analysis easier
- The covariance matrix of $\hat{\beta}$ is then

$$\begin{aligned}\mathrm{Var}(\hat{\beta}) = \mathrm{Var}((\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t y) &= (\mathbf{X}^t\mathbf{X})^{-1} X^t \,\mathrm{Var}(y)\, X (\mathbf{X}^t\mathbf{X})^{-1} \\ &= (\mathbf{X}^t\mathbf{X})^{-1}\sigma^2\end{aligned}$$

- Usually one estimates the variance σ^2 with

$$\hat{\sigma}^2 = \frac{1}{n-p-1}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Analysis of the distribution of $\hat{\beta}$

- To say more we need to make more assumptions. Therefore assume

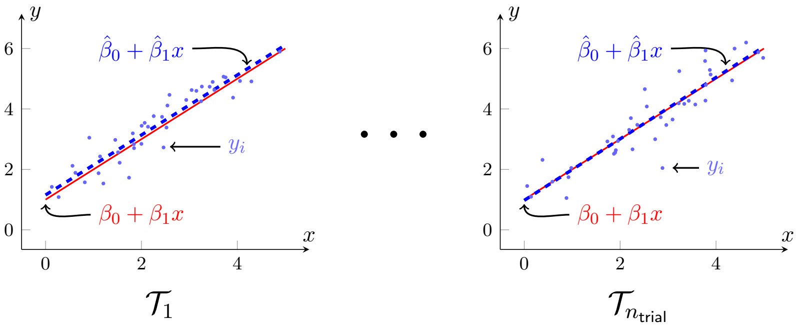
$$\begin{aligned} Y &= \mathrm{E}(Y \mid X_1, X_2, \ldots, X_p) + \epsilon \\ &= \beta_0 + \sum_{i=1}^{p} X_j \beta_j + \epsilon \end{aligned}$$

  where $\epsilon \sim N(0, \sigma^2)$

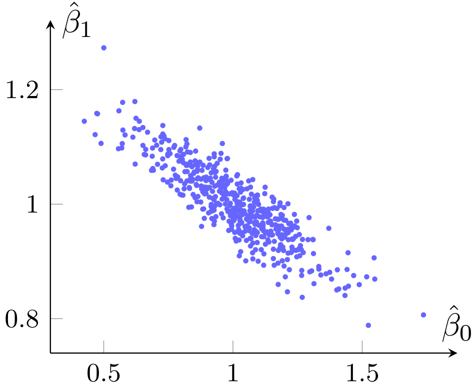
- Then it's easy to show that (assuming non-random x_i)

$$\hat{\beta} \sim N(\beta, (\mathbf{X}^t \mathbf{X})^{-1} \sigma^2)$$

Given this additive model generate $\hat{\beta}$

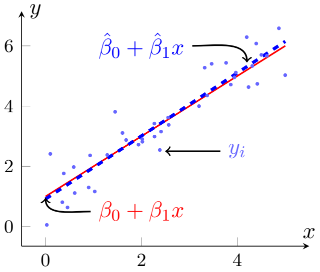


- $\mathcal{T}$ is a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- $\beta = (1,1)^t, n = 40, \sigma = .6$
- In this simulation the x_i's differ across trials.



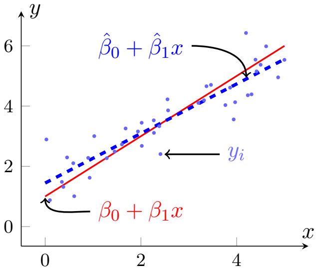
Each $\mathcal{T}_i$ results in a different estimate of $\hat{\beta}$. Have plotted these $\hat{\beta}$'s for $n_{\text{trial}} = 500$.

Given this additive model generate $\hat{\beta}$



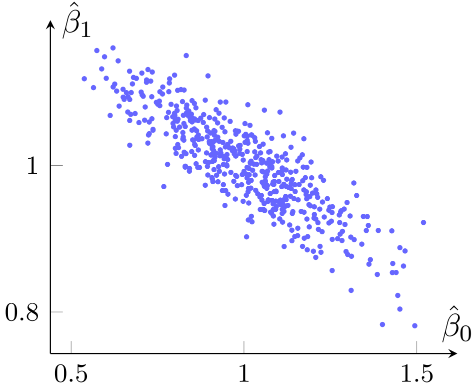
$\mathcal{T}_1$

. . .



$\mathcal{T}_{n_{\text{trial}}}$

- $\mathcal{T}$ is a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- $\beta = (1, 1)^t, n = 40, \sigma = .6$
- In this simulation the x_i's are fixed across trials.



Each $\mathcal{T}_i$ results in a different estimate of $\hat{\beta}$. Have plotted these $\hat{\beta}$'s for $n_{\mathsf{trial}} = 500$.

Which β_j's are probably zero?

- To interpret the weights estimated by least squares it would be nice to say which ones are probably zero.

- The associated predictors can then be removed from the model.

- If $\beta_j = 0$ then $\hat{\beta} \sim N(0, \sigma^2 v_{jj})$ where v_{jj} is the jth diagonal element of $(\mathbf{X}^t\mathbf{X})^{-1}$.

- Then if the actual value computed for $\hat{\beta}_j$ is larger than $\sigma^2 v_{jj}$ then it is highly improbable that $\beta_j = 0$.

- Statisticians have exact tests based on suitable distributions. In this case compute

$$z_j = \frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{v_{jj}}}$$

  and if $\beta_j = 0$ then z_j has a t-distribution with $n - p - 1$ dof.

Which β_j's are probably zero?

- To interpret the weights estimated by least squares it would be nice to say which ones are probably zero.
- The associated predictors can then be removed from the model.
- If $\beta_j = 0$ then $\hat{\beta} \sim N(0, \sigma^2 v_{jj})$ where v_{jj} is the jth diagonal element of $(\mathbf{X}^t\mathbf{X})^{-1}$.
- Then if the actual value computed for $\hat{\beta}_j$ is larger than $\sigma^2 v_{jj}$ then it is highly improbable that $\beta_j = 0$.
- Statisticians have exact tests based on suitable distributions. In this case compute

$$z_j = \frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{v_{jj}}}$$

and if $\beta_j = 0$ then z_j has a t-distribution with $n - p - 1$ dof.

Which β_j's are probably zero?

- To interpret the weights estimated by least squares it would be nice to say which ones are probably zero.
- The associated predictors can then be removed from the model.
- If $\beta_j = 0$ then $\hat{\beta} \sim N(0, \sigma^2 v_{jj})$ where v_{jj} is the jth diagonal element of $(\mathbf{X}^t\mathbf{X})^{-1}$.
- Then if the actual value computed for $\hat{\beta}_j$ is larger than $\sigma^2 v_{jj}$ then it is highly improbable that $\beta_j = 0$.
- Statisticians have exact tests based on suitable distributions. In this case compute

$$z_j = \frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{v_{jj}}}$$

and if $\beta_j = 0$ then z_j has a t-distribution with $n - p - 1$ dof.

Which β_j's are probably zero?

- To interpret the weights estimated by least squares it would be nice to say which ones are probably zero.

- The associated predictors can then be removed from the model.

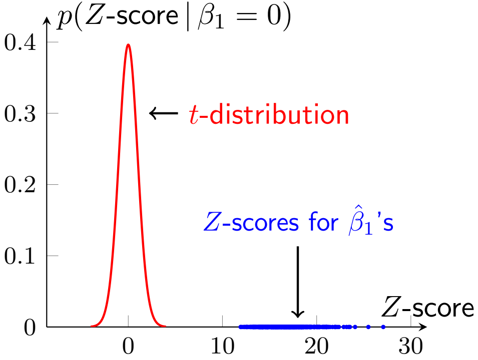
- If $\beta_j = 0$ then $\hat{\beta} \sim N(0, \sigma^2 v_{jj})$ where v_{jj} is the jth diagonal element of $(\mathbf{X}^t\mathbf{X})^{-1}$.

- Then if the actual value computed for $\hat{\beta}_j$ is larger than $\sigma^2 v_{jj}$ then it is highly improbable that $\beta_j = 0$.

- Statisticians have exact tests based on suitable distributions. In this case compute

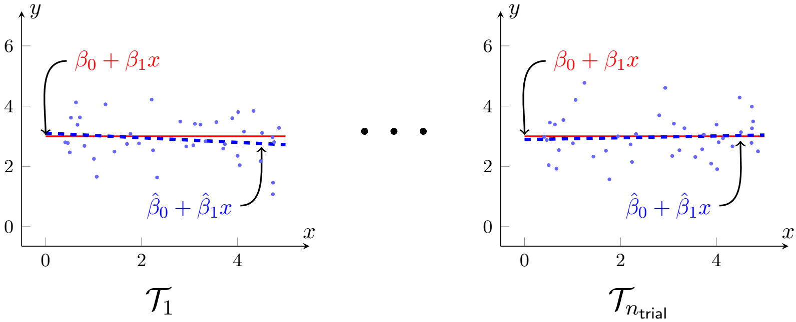
$$z_j = \frac{\hat{\beta}_j}{\hat{\sigma}\sqrt{v_{jj}}}$$

and if $\beta_j = 0$ then z_j has a t-distribution with $n - p - 1$ dof.

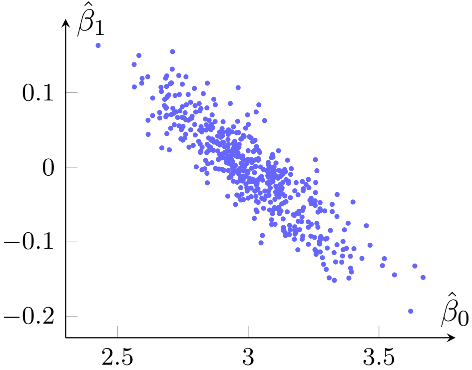


- For the example we had with $\beta = (1,1)^t, n = 40$ and $\sigma = .6$ then the t-distribution of z_1 is shown if $\beta_j = 0$.
- The z_1 computed from each $\hat{\beta}$ estimated with $\mathcal{T}_i$ is shown.
- Obviously even if we didn't know $\hat{\beta}$ and only saw one $\mathcal{T}_i$ we would not think $\beta_j \neq 0$.

Look at an example when $\beta_1 = 0$

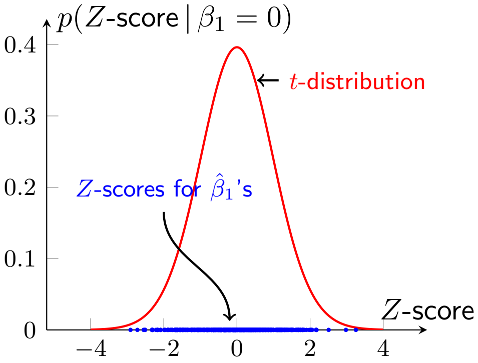


- $\mathcal{T}$ is a training set $\{(x_i, y_i)\}_{i=1}^n$
- $\beta = (3, 0)^t, n = 40, \sigma = .6$
- In this simulation the x_i's are fixed across trials.



Each $\mathcal{T}_i$ results in a different estimate of $\hat{\beta}$. Have plotted these $\hat{\beta}$'s for $n_{\text{trial}} = 500$.

Is β_1 zero?



- For this example we have $\beta = (3, 0)^t, n = 40$ and $\sigma = .6$ then the t-distribution of z_1 is shown if $\beta_j = 0$
- The z_1's computed from the $\hat{\beta}$ estimated with $\mathcal{T}_i$ are shown.
- Obviously even if we didn't know $\hat{\beta}$ and only saw one $\mathcal{T}_i$ we would conclude in most trials that $\beta_j \neq 0$.

Other tests can be performed

We will not look into these but you can

- test for the significance of groups of coefficients simultaneously
- get confidence bounds for β_j centred at $\hat{\beta}_j$.

Gauss-Markov Theorem

- A famous result in statistics

  The least squares estimate $\hat{\beta}^{ls}$ of the parameters β has the smallest variance among all linear unbiased estimates.

- To explain a simple case of the theorem. Let $\theta = a^t\beta$.

- The least squares estimate of $a^t\beta$ is

$$\hat{\theta} = a^t\hat{\beta}^{\mathsf{ls}} = a^t(\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t y$$

  If $\mathbf{X}$ is fixed this is a linear function, $c_0^t y$, of the response vector y.

- If we assume $\mathrm{E}[y] = X\beta$ then $a^t\hat{\beta}^{\mathsf{ls}}$ is unbiased

$$\mathrm{E}[a^t\hat{\beta}^{\mathsf{ls}}] = \mathrm{E}[a^t(\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t y] = a^t(\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t\mathbf{X}\beta = a^t\beta = \theta$$

Gauss-Markov Theorem: Simple example

- **Gauss-Markov Theorem** states any other linear estimator $\tilde{\theta} = c^t y$ that is unbiased for $a^t\beta$ has

$$\mathrm{Var}[a^t \hat{\beta}^{\mathsf{ls}}] \le \mathrm{Var}[c^t y]$$

- Have only stated the result for the estimation of one parameter $a^t\beta$ but can state it in terms of the entire parameter vector β.

- However, having an unbiased estimator is not always crucial.

Gauss-Markov Theorem: Simple example

- **Gauss-Markov Theorem** states any other linear estimator $\tilde{\theta} = c^t y$ that is unbiased for $a^t \beta$ has

$$\text{Var}[a^t \hat{\beta}^{\text{ls}}] \leq \text{Var}[c^t y]$$

- Have only stated the result for the estimation of one parameter $a^t \beta$ but can state it in terms of the entire parameter vector β.

- However, having an unbiased estimator is not always crucial.

Gauss-Markov Theorem: Simple example

- **Gauss-Markov Theorem** states any other linear estimator $\tilde{\theta} = c^t y$ that is unbiased for $a^t \beta$ has

$$\mathrm{Var}[a^t \hat{\beta}^{\mathsf{ls}}] \leq \mathrm{Var}[c^t y]$$

- Have only stated the result for the estimation of one parameter $a^t \beta$ but can state it in terms of the entire parameter vector β.

- However, having an unbiased estimator is not always crucial.

The Bias - Variance Trade-off (once again!)

- Consider the mean-squared error of an estimator $\tilde{\theta}$ in estimating θ

$$\begin{aligned}\mathrm{MSE}(\tilde{\theta}) &= \mathrm{E}((\tilde{\theta}-\theta)^2) \\ &= \underbrace{\mathrm{Var}(\tilde{\theta})}_{\text{variance}} + (\underbrace{\mathrm{E}(\tilde{\theta}) - \theta}_{\text{bias}})^2\end{aligned}$$

- **Gauss-Markov** says the least square estimator has the smallest MSE for all linear estimators with zero bias.

- But there may be biased estimates with smaller MSE.

- In these cases have traded an increase in squared bias for a reduction in variance.

The Bias - Variance Trade-off (once again!)

- Consider the mean-squared error of an estimator $\tilde{\theta}$ in estimating θ

$$\begin{aligned}\mathrm{MSE}(\tilde{\theta}) &= \mathrm{E}((\tilde{\theta}-\theta)^2) \\ &= \underbrace{\mathrm{Var}(\tilde{\theta})}_{\text{variance}} + (\underbrace{\mathrm{E}(\tilde{\theta}) - \theta}_{\text{bias}})^2\end{aligned}$$

- **Gauss-Markov** says the least square estimator has the smallest MSE for all linear estimators with zero bias.

- But there may be biased estimates with smaller MSE.

- In these cases have traded an increase in squared bias for a reduction in variance.

The Bias - Variance Trade-off (once again!)

- Consider the mean-squared error of an estimator $\tilde{\theta}$ in estimating θ

$$\begin{aligned}\mathrm{MSE}(\tilde{\theta}) &= \mathrm{E}((\tilde{\theta}-\theta)^2) \\ &= \underbrace{\mathrm{Var}(\tilde{\theta})}_{\text{variance}} + (\underbrace{\mathrm{E}(\tilde{\theta})-\theta}_{\text{bias}})^2\end{aligned}$$

- **Gauss-Markov** says the least square estimator has the smallest MSE for all linear estimators with zero bias.
- But there may be biased estimates with smaller MSE.
- In these cases have traded an increase in squared bias for a reduction in variance.

Simple Univariate Regression and *Gram-Schmidt*

Multiple regression from simple univariate regression

- Suppose we have univariate model with no intercept

$$Y = X\beta + \epsilon$$

- The least square estimate is

$$\hat{\beta} = \frac{\langle x, y \rangle}{\langle x, x \rangle}$$

where $x = (x_1, x_2, \ldots, x_n)^t$ and $y = (y_1, y_2, \ldots, y_n)$.

- The residuals are given by

$$r = y - x^t \hat{\beta}$$

- Say $x_i \in \mathbb{R}^p$ and the columns of $\mathbf{X}$ are orthogonal then

$$\hat{\beta}_j = \frac{\langle x_{.j}, y \rangle}{\langle x_{.j}, x_{.j} \rangle}, \qquad \text{where } x_{.j} \text{ is } j\text{th column of } \mathbf{X}$$

Multiple regression from simple univariate regression

- Suppose we have univariate model with no intercept

$$Y = X\beta + \epsilon$$

- The least square estimate is

$$\hat{\beta} = \frac{\langle x, y \rangle}{\langle x, x \rangle}$$

where $x = (x_1, x_2, \ldots, x_n)^t$ and $y = (y_1, y_2, \ldots, y_n)$.

- The residuals are given by

$$r = y - x^t \hat{\beta}$$

- Say $x_i \in \mathbb{R}^p$ and the columns of $\mathbf{X}$ are orthogonal then

$$\hat{\beta}_j = \frac{\langle x_{.j}, y \rangle}{\langle x_{.j}, x_{.j} \rangle}, \qquad \text{where } x_{.j} \text{ is } j\text{th column of } \mathbf{X}$$

Multiple regression from simple univariate regression

- Suppose we have univariate model with no intercept

$$Y = X\beta + \epsilon$$

- The least square estimate is

$$\hat{\beta} = \frac{\langle x, y \rangle}{\langle x, x \rangle}$$

where $x = (x_1, x_2, \ldots, x_n)^t$ and $y = (y_1, y_2, \ldots, y_n)$.

- The residuals are given by

$$r = y - x^t \hat{\beta}$$

- Say $x_i \in \mathbb{R}^p$ and the columns of $\mathbf{X}$ are orthogonal then

$$\hat{\beta}_j = \frac{\langle x_{.j}, y \rangle}{\langle x_{.j}, x_{.j} \rangle}, \qquad \text{where } x_{.j} \text{ is } j\text{th column of } \mathbf{X}$$

Multiple regression from simple univariate regression

- Suppose we have univariate model with no intercept

$$Y = X\beta + \epsilon$$

- The least square estimate is

$$\hat{\beta} = \frac{\langle x, y \rangle}{\langle x, x \rangle}$$

where $x = (x_1, x_2, \ldots, x_n)^t$ and $y = (y_1, y_2, \ldots, y_n)$.

- The residuals are given by

$$r = y - x^t \hat{\beta}$$

- Say $x_i \in \mathbb{R}^p$ and the columns of $\mathbf{X}$ are orthogonal then

$$\hat{\beta}_j = \frac{\langle x_{.j}, y \rangle}{\langle x_{.j}, x_{.j} \rangle}, \qquad \text{where } x_{.j} \text{ is } j\text{th column of } \mathbf{X}$$

OLS via successive orthogonalization

- $\mathbf{X}$ acquired from observations are rarely orthogonal.
- Hence they have to be orthogonalized to take advantage the previous insight.
- $x_{.0}$ be the 0th column of $\mathbf{X} \in \mathbb{R}^{n \times 2}$ (vector of ones) then
  - Regress $x_{.1}$ on $x_{.0}$ that is $\hat{\gamma} = \dfrac{\langle x_{.0}, x_{.1} \rangle}{\langle x_{.0}, x_{.0} \rangle}$ and let $z = x_{.1} - \hat{\gamma}\, x_{.0}$
  - Regress y on z then $\hat{\beta}_1 = \dfrac{\langle x_{.1}, z \rangle}{\langle x_{.1}, x_{.1} \rangle}$
  - Then $y \approx \hat{\beta}_1 z = \hat{\beta}_1(x_{.1} - \hat{\gamma} x_{.0}) = \mathbf{X}\hat{\beta}$ where $\hat{\beta} = (\hat{\beta}_1, -\hat{\beta}_1\hat{\gamma})^t$. The solution is same as if one had directly calculated $\hat{\beta}^{\mathrm{ls}}$. Have just used an orthogonal basis for the col. space of $\mathbf{X}$
- Note Step 1 **orthogonalized** $x_{.1}$ w.r.t. $x_{.0}$.
- Step 2 is simple univariate regression using the orthogonal predictors $x_{.0}$ and z.

OLS via successive orthogonalization

- $\mathbf{X}$ acquired from observations are rarely orthogonal.
- Hence they have to be orthogonalized to take advantage the previous insight.
- $x_{.0}$ be the 0th column of $\mathbf{X} \in \mathbb{R}^{n \times 2}$ (vector of ones) then
    - Regress $x_{.1}$ on $x_{.0}$ that is $\hat{\gamma} = \dfrac{\langle x_{.0}, x_{.1} \rangle}{\langle x_{.0}, x_{.0} \rangle}$ and let $z = x_{.1} - \hat{\gamma}\, x_{.0}$
    - Regress y on z then $\hat{\beta}_1 = \dfrac{\langle x_{.1}, z \rangle}{\langle x_{.1}, x_{.1} \rangle}$
    - Then $y \approx \hat{\beta}_1 z = \hat{\beta}_1 (x_{.1} - \hat{\gamma} x_{.0}) = \mathbf{X}\hat{\beta}$ where $\hat{\beta} = (\hat{\beta}_1, -\hat{\beta}_1 \hat{\gamma})^t$. The solution is same as if one had directly calculated $\hat{\beta}^{\mathsf{ls}}$. Have just used an orthogonal basis for the col. space of $\mathbf{X}$
- Note Step 1 **orthogonalized** $x_{.1}$ w.r.t. $x_{.0}$.
- Step 2 is simple univariate regression using the orthogonal predictors $x_{.0}$ and z.

OLS via successive orthogonalization

- $\mathbf{X}$ acquired from observations are rarely orthogonal.
- Hence they have to be orthogonalized to take advantage the previous insight.
- $x_{.0}$ be the 0th column of $\mathbf{X} \in \mathbb{R}^{n \times 2}$ (vector of ones) then
  - Regress $x_{.1}$ on $x_{.0}$ that is $\hat{\gamma} = \dfrac{\langle x_{.0}, x_{.1} \rangle}{\langle x_{.0}, x_{.0} \rangle}$ and let $z = x_{.1} - \hat{\gamma}\, x_{.0}$
  - Regress y on z then $\hat{\beta}_1 = \dfrac{\langle x_{.1}, z \rangle}{\langle x_{.1}, x_{.1} \rangle}$
  - Then $y \approx \hat{\beta}_1 z = \hat{\beta}_1 (x_{.1} - \hat{\gamma} x_{.0}) = \mathbf{X}\hat{\beta}$ where $\hat{\beta} = (\hat{\beta}_1, -\hat{\beta}_1 \hat{\gamma})^t$. The solution is same as if one had directly calculated $\hat{\beta}^{\mathsf{ls}}$. Have just used an orthogonal basis for the col. space of $\mathbf{X}$
- Note Step 1 **orthogonalized** $x_{.1}$ w.r.t. $x_{.0}$.
- Step 2 is simple univariate regression using the orthogonal predictors $x_{.0}$ and z.

OLS via successive orthogonalization

- $\mathbf{X}$ acquired from observations are rarely orthogonal.
- Hence they have to be orthogonalized to take advantage the previous insight.
- $x_{.0}$ be the 0th column of $\mathbf{X} \in \mathbb{R}^{n\times 2}$ (vector of ones) then
  - Regress $x_{.1}$ on $x_{.0}$ that is $\hat{\gamma} = \dfrac{\langle x_{.0}, x_{.1}\rangle}{\langle x_{.0}, x_{.0}\rangle}$ and let $z = x_{.1} - \hat{\gamma}\, x_{.0}$
  - Regress y on z then $\hat{\beta}_1 = \dfrac{\langle x_{.1}, z\rangle}{\langle x_{.1}, x_{.1}\rangle}$
  - Then $y \approx \hat{\beta}_1 z = \hat{\beta}_1(x_{.1} - \hat{\gamma} x_{.0}) = \mathbf{X}\hat{\beta}$ where $\hat{\beta} = (\hat{\beta}_1, -\hat{\beta}_1\hat{\gamma})^t$. The solution is same as if one had directly calculated $\hat{\beta}^{\mathsf{ls}}$. Have just used an orthogonal basis for the col. space of $\mathbf{X}$
- Note Step 1 **orthogonalized** $x_{.1}$ w.r.t. $x_{.0}$.
- Step 2 is simple univariate regression using the orthogonal predictors $x_{.0}$ and z.

OLS via successive orthogonalization

- Can extend the process to when x_i's are p-dimensional.
- See Algorithm 3.1 in the book.
- At each iteration j a multiple least squares regression problem with jth orthogonal inputs is solved.
- And after this a new residual is formed which is orthogonal to all these current directions.
- This process is the **Gram-Schmidt** regression procedure.

OLS via successive orthogonalization

- Can extend the process to when x_i's are p-dimensional.
- See Algorithm 3.1 in the book.
- At each iteration j a multiple least squares regression problem with jth orthogonal inputs is solved.
- And after this a new residual is formed which is orthogonal to all these current directions.
- This process is the **Gram-Schmidt** regression procedure.

Subset Selection

Inadequacies of least squares estimates

- **Prediction Accuracy**
  Least squares estimates often have
  - Low bias **and** high variance
  - This can affect prediction accuracy
- Frequently better to set some of the β_j's to zero.
- This increases the bias but reduces the variance and in turn improve prediction accuracy.
- **Interpretation**
  For p large, it may be difficult to decipher the important factors.
- Therefore would like to determine a smaller subset of predictors which are most informative.
  May sacrifice *small detail* for the *big picture*.

Inadequacies of least squares estimates

- **Prediction Accuracy**
  Least squares estimates often have
  - Low bias **and** high variance
  - This can affect prediction accuracy
- Frequently better to set some of the β_j's to zero.
- This increases the bias but reduces the variance and in turn improve prediction accuracy.
- **Interpretation**
  For p large, it may be difficult to decipher the important factors.
- Therefore would like to determine a smaller subset of predictors which are most informative.
  May sacrifice *small detail* for the *big picture*.

Inadequacies of least squares estimates

- **Prediction Accuracy**
  Least squares estimates often have
  - Low bias **and** high variance
  - This can affect prediction accuracy
- Frequently better to set some of the β_j's to zero.
- This increases the bias but reduces the variance and in turn improve prediction accuracy.
- **Interpretation**
  For p large, it may be difficult to decipher the important factors.
- Therefore would like to determine a smaller subset of predictors which are most informative.
  May sacrifice *small detail* for the *big picture*.

Inadequacies of least squares estimates

- **Prediction Accuracy**
  Least squares estimates often have
  - Low bias **and** high variance
  - This can affect prediction accuracy
- Frequently better to set some of the β_j's to zero.
- This increases the bias but reduces the variance and in turn improve prediction accuracy.
- **Interpretation**
  For p large, it may be difficult to decipher the important factors.
- Therefore would like to determine a smaller subset of predictors which are most informative.
  May sacrifice *small detail* for the *big picture*.

Inadequacies of least squares estimates

- **Prediction Accuracy**
  Least squares estimates often have
  - Low bias **and** high variance
  - This can affect prediction accuracy
- Frequently better to set some of the β_j's to zero.
- This increases the bias but reduces the variance and in turn improve prediction accuracy.
- **Interpretation**
  For p large, it may be difficult to decipher the important factors.
- Therefore would like to determine a smaller subset of predictors which are most informative.
  May sacrifice *small detail* for the *big picture*.

- **Best subset regression** finds for $k \in \{0, 1, 2, \ldots, p\}$ the $j_1, j_2, \ldots, j_k$ with each $j_l \in \{1, 2, \ldots, p\}$ s.t.

$$\mathrm{RSS}(j_1, j_2, \ldots, j_k) = \min_{\beta_0, \beta_{j_1}, \ldots, \beta_{j_l}} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{l=1}^{k} \beta_{j_l} x_{i,j_l})^2$$

is smallest.

- There are $\binom{p}{k}$ different subsets to try for a given k.
- If $p \leq 40$ there exist computational feasible algorithms for finding these best subsets of size k.
- Question still remains of how to choose best value of k.
- Once again it is a trade-off between bias and variance....

- **Best subset regression** finds for $k \in \{0, 1, 2, \ldots, p\}$ the $j_1, j_2, \ldots, j_k$ with each $j_l \in \{1, 2, \ldots, p\}$ s.t.

$$\mathrm{RSS}(j_1, j_2, \ldots, j_k) = \min_{\beta_0, \beta_{j_1}, \ldots, \beta_{j_l}} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{l=1}^{k} \beta_{j_l} x_{i,j_l})^2$$

  is smallest.

- There are $\binom{p}{k}$ different subsets to try for a given k.

- If $p \leq 40$ there exist computational feasible algorithms for finding these best subsets of size k.

- Question still remains of how to choose best value of k.

- Once again it is a trade-off between bias and variance....

- **Best subset regression** finds for $k \in \{0, 1, 2, \ldots, p\}$ the $j_1, j_2, \ldots, j_k$ with each $j_l \in \{1, 2, \ldots, p\}$ s.t.

$$\mathrm{RSS}(j_1, j_2, \ldots, j_k) = \min_{\beta_0, \beta_{j_1}, \ldots, \beta_{j_l}} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{l=1}^{k} \beta_{j_l} x_{i,j_l})^2$$

  is smallest.

- There are $\binom{p}{k}$ different subsets to try for a given k.

- If $p \leq 40$ there exist computational feasible algorithms for finding these best subsets of size k.

- Question still remains of how to choose best value of k.

- Once again it is a trade-off between bias and variance....

- **Best subset regression** finds for $k \in \{0, 1, 2, \ldots, p\}$ the $j_1, j_2, \ldots, j_k$ with each $j_l \in \{1, 2, \ldots, p\}$ s.t.

$$\text{RSS}(j_1, j_2, \ldots, j_k) = \min_{\beta_0, \beta_{j_1}, \ldots, \beta_{j_l}} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{l=1}^{k} \beta_{j_l} x_{i,j_l})^2$$

  is smallest.

- There are $\binom{p}{k}$ different subsets to try for a given k.

- If $p \leq 40$ there exist computational feasible algorithms for finding these best subsets of size k.

- Question still remains of how to choose best value of k.

- Once again it is a trade-off between bias and variance....

Best-Subset Selection

- **Best subset regression** finds for $k \in \{0, 1, 2, \ldots, p\}$ the $j_1, j_2, \ldots, j_k$ with each $j_l \in \{1, 2, \ldots, p\}$ s.t.

$$\mathrm{RSS}(j_1, j_2, \ldots, j_k) = \min_{\beta_0, \beta_{j_1}, \ldots, \beta_{j_l}} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{l=1}^{k} \beta_{j_l} x_{i,j_l})^2$$

  is smallest.

- There are $\binom{p}{k}$ different subsets to try for a given k.

- If $p \leq 40$ there exist computational feasible algorithms for finding these best subsets of size k.

- Question still remains of how to choose best value of k.

- Once again it is a trade-off between bias and variance....

- Instead of searching all possible subsets (infeasible for large p) can take a greedy approach.

- The steps of **Forward-Stepwise Selection** are
  - Set $\mathcal{I} = \{1, \ldots, p\}$
  - For $l = 1, \ldots, k$ choose j_l according to

$$j_l = \arg\min_{j \in \mathcal{I}} \min_{\beta_0, \beta_{j_1}, \ldots, \beta_{j_{l-1}}, \beta_j} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{s=1}^{l-1} \beta_{j_s} x_{ij_s} - \beta_j x_{ij})^2$$

and

$$\mathcal{I} = \mathcal{I} \backslash \{j_l\}$$

- Instead of searching all possible subsets (infeasible for large p) can take a greedy approach.
- The steps of **Forward-Stepwise Selection** are
  - Set $\mathcal{I} = \{1, \ldots, p\}$
  - For $l = 1, \ldots, k$ choose j_l according to

$$j_l = \arg\min_{j \in \mathcal{I}} \min_{\beta_0, \beta_{j_1}, \ldots, \beta_{j_{l-1}}, \beta_j} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{s=1}^{l-1} \beta_{j_s} x_{ij_s} - \beta_j x_{ij})^2$$

and

$$\mathcal{I} = \mathcal{I} \backslash \{j_l\}$$

Forward-Stepwise Vs Best Subset Selection

- Forward-Stepwise may be sub-optimal compared to the best subset selection but may be preferred because
  - It is computational feasible for large $p \gg n$. Not true for best subset selection.
  - best subset selection may overfit.
  - Forward stepwise will probably produce a function with lower variance but perhaps more bias.

Forward-Stepwise Vs Best Subset Selection

- Forward-Stepwise may be sub-optimal compared to the best subset selection but may be preferred because
  - It is computational feasible for large $p \gg n$. Not true for best subset selection.
  - best subset selection may overfit.
  - Forward stepwise will probably produce a function with lower variance but perhaps more bias.

Forward-Stepwise Vs Best Subset Selection

- Forward-Stepwise may be sub-optimal compared to the best subset selection but may be preferred because
  - It is computational feasible for large $p \gg n$. Not true for best subset selection.
  - best subset selection may overfit.
  - Forward stepwise will probably produce a function with lower variance but perhaps more bias.

Forward-Stepwise Vs Best Subset Selection

- Forward-Stepwise may be sub-optimal compared to the best subset selection but may be preferred because
  - It is computational feasible for large $p \gg n$. Not true for best subset selection.
  - best subset selection may overfit.
  - Forward stepwise will probably produce a function with lower variance but perhaps more bias.

- The steps of **Forward-Stagewise Regression** are
  - Set $\hat{\beta}_0 = \dfrac{1}{n}\sum_{i=1}^{n} y_i$
  - Set $\hat{\beta}_1 = \hat{\beta}_2 = \cdots = \hat{\beta}_p = 0$
  - At each iteration

$$r_i = y_i - \hat{\beta}_0 - \sum_{j=1}^{p} \hat{\beta}_j x_{ij}, \quad \text{compute residual for each example}$$

$$j^* = \arg\max_{j \in \mathcal{I}} |\langle x_{\cdot j}, r\rangle| \quad \text{find } X_j \text{ most correlated with } r$$

$$\hat{\beta}_{j^*} \leftarrow \hat{\beta}_{j^*} + \delta\, \mathrm{sign}(\langle x_{\cdot j^*}, r\rangle)$$

  - Stop iterations when the residuals are uncorrelated with all the predictors.

Forward-Stagewise Regression: What to note..

- Only one $\hat{\beta}_j$ is updated at each iteration.
- A $\hat{\beta}_j$ can be updated at several different iterations.
- It can be slow to reach the least squares fit.
- But slow fitting may not be such a bad thing in high dimensional problems.

Forward-Stagewise Regression: What to note..

- Only one $\hat{\beta}_j$ is updated at each iteration.
- A $\hat{\beta}_j$ can be updated at several different iterations.
- It can be slow to reach the least squares fit.
- But slow fitting may not be such a bad thing in high dimensional problems.

Forward-Stagewise Regression: What to note..

- Only one $\hat{\beta}_j$ is updated at each iteration.
- A $\hat{\beta}_j$ can be updated at several different iterations.
- It can be slow to reach the least squares fit.
- But slow fitting may not be such a bad thing in high dimensional problems.

Forward-Stagewise Regression: What to note..

- Only one $\hat{\beta}_j$ is updated at each iteration.
- A $\hat{\beta}_j$ can be updated at several different iterations.
- It can be slow to reach the least squares fit.
- But slow fitting may not be such a bad thing in high dimensional problems.

Shrinkage methods

Why shrinkage methods?

- Selecting a subset of predictors produces a model that is interpretable and probably has lower prediction error than the full model.
- **However** it is a discrete process $\implies$ introduces variation into learning the model.
- Shrinkage methods are more continuous and have a lower variance.

Why shrinkage methods?

- Selecting a subset of predictors produces a model that is interpretable and probably has lower prediction error than the full model.
- **However** it is a discrete process $\implies$ introduces variation into learning the model.
- Shrinkage methods are more continuous and have a lower variance.

- Selecting a subset of predictors produces a model that is interpretable and probably has lower prediction error than the full model.

- **However** it is a discrete process $\implies$ introduces variation into learning the model.

- Shrinkage methods are more continuous and have a lower variance.

Ridge Regression

Shrinkage Method 1: Ridge Regression

- **Ridge regression** shrinks β_j's by imposing a penalty on their size.

- The ridge coefficients minimize a penalized RSS

non-negative complexity parameter

$$\hat{\beta}^{\text{ridge}} = \arg\min_{\beta} \left\{ \underbrace{\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2}_{\text{residual sum-of-squares}} + \lambda \underbrace{\sum_{j=1}^{p}\beta_j^2}_{\text{regularizer fn}} \right\}$$

- The larger $\lambda \geq 0$ the greater of the amount of shrinkage. This implies β_j's are shrunk toward zero (except β_0).

Shrinkage Method 1: Ridge Regression

- **Ridge regression** shrinks β_j's by imposing a penalty on their size.

- The ridge coefficients minimize a penalized RSS

non-negative complexity parameter

$$\hat{\beta}^{\text{ridge}} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

residual sum-of-squares

regularizer fn

- The larger $\lambda \geq 0$ the greater of the amount of shrinkage. This implies β_j's are shrunk toward zero (except β_0).

Shrinkage Method 1: Ridge Regression

- **Ridge regression** shrinks β_j's by imposing a penalty on their size.

- The ridge coefficients minimize a penalized RSS

non-negative complexity parameter

$$\hat{\beta}^{\text{ridge}} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

residual sum-of-squares

regularizer fn

- The larger $\lambda \geq 0$ the greater of the amount of shrinkage. This implies β_j's are shrunk toward zero (except β_0).

An equivalent formulation of Ridge Regression

$$\hat{\beta}^{\text{ridge}} = \arg\min_{\beta} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 \leq t$$

- This formulation puts an explicit constraint on the size of the β_j's.
- There is a 1-1 correspondence between λ and t in the two formulations.
- Note the estimated $\hat{\beta}^{\text{ridge}}$ changes if the scaling of the inputs change.

Ridge Regression and centering of data

- The centered version of the input data is

$$\tilde{x}_{ij} = x_{ij} - \sum_{s=1}^{n} x_{sj}$$

Then the ridge regression coefficients found using the centered data

$$\hat{\beta}^c = \arg\min_{\beta^c} \sum_{i=1}^{n} (y_i - \beta_0^c - \sum_{j=1}^{p} \tilde{x}_{ij}\beta_j^c)^2 + \lambda \sum_{j=1}^{p} (\beta_j^c)^2$$

are related to the coefficients found using the original data via

$$\hat{\beta}_0^c = \frac{1}{n}\sum_{i=1}^{n} y_i = \bar{y}, \qquad \hat{\beta}_0^{\text{ridge}} = \bar{y} - \sum_{j=1}^{p} \bar{x}_{.j}\hat{\beta}_j^{\text{ridge}}$$

$$\hat{\beta}_j^c = \hat{\beta}_j^{\text{ridge}} \quad \text{for } i = 1, \ldots, p$$

Ridge Regression and centering of data

- If the y's have zero mean $\implies \hat{\beta}_0^c = 0$

- Can drop the intercept term from the linear model if the input data is centred.

- Then for ridge regression, given all the necessary centering, find the $\beta = (\beta_1, \ldots, \beta_p)^t$ which minimizes

$$\begin{aligned}\hat{\beta} &= \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} \\ &= \arg\min_{\beta} \left\{ (y - \mathbf{X}\beta)^t (y - \mathbf{X}\beta) + \lambda \beta^t \beta \right\}\end{aligned}$$

where $y = (y_1, \ldots, y_n)^t$ and

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

Ridge Regression and centering of data

- If the y's have zero mean $\implies \hat{\beta}_0^c = 0$
- Can drop the intercept term from the linear model if the input data is centred.
- Then for ridge regression, given all the necessary centering, find the $\beta = (\beta_1, \ldots, \beta_p)^t$ which minimizes

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

$$= \arg\min_{\beta} \left\{ (y - \mathbf{X}\beta)^t (y - \mathbf{X}\beta) + \lambda \beta^t \beta \right\}$$

where $y = (y_1, \ldots, y_n)^t$ and

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

Ridge Regression and centering of data

- If the y's have zero mean $\implies \hat{\beta}_0^c = 0$
- Can drop the intercept term from the linear model if the input data is centred.
- Then for ridge regression, given all the necessary centering, find the $\beta = (\beta_1, \ldots, \beta_p)^t$ which minimizes

$$\begin{aligned}\hat{\beta} &= \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} \\ &= \arg\min_{\beta} \left\{ (y - \mathbf{X}\beta)^t (y - \mathbf{X}\beta) + \lambda \beta^t \beta \right\}\end{aligned}$$

where $y = (y_1, \ldots, y_n)^t$ and

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

Ridge Regression and centering of data

- For rest of lecture will assume centered input and output data.
- The ridge regression solution is given by

$$\hat{\beta}^{\text{ridge}} = (\mathbf{X}^t\mathbf{X} + \lambda I_p)^{-1}\mathbf{X}^t y$$

- Note that the problem of inverting the potentially singular matrix $\mathbf{X}^t\mathbf{X}$ is averted as $(\mathbf{X}^t\mathbf{X} + \lambda\mathbf{I}_p)$ is full rank even if $\mathbf{X}^t\mathbf{X}$ is not.

Ridge Regression and centering of data

- For rest of lecture will assume centered input and output data.

- The ridge regression solution is given by

$$\hat{\beta}^{\text{ridge}} = (\mathbf{X}^t\mathbf{X} + \lambda I_p)^{-1}\mathbf{X}^t y$$

- Note that the problem of inverting the potentially singular matrix $\mathbf{X}^t\mathbf{X}$ is averted as $(\mathbf{X}^t\mathbf{X} + \lambda\mathbf{I}_p)$ is full rank even if $\mathbf{X}^t\mathbf{X}$ is not.

Ridge Regression and centering of data

- For rest of lecture will assume centered input and output data.
- The ridge regression solution is given by

$$\hat{\beta}^{\text{ridge}} = (\mathbf{X}^t\mathbf{X} + \lambda I_p)^{-1}\mathbf{X}^t y$$

- Note that the problem of inverting the potentially singular matrix $\mathbf{X}^t\mathbf{X}$ is averted as $(\mathbf{X}^t\mathbf{X} + \lambda\mathbf{I}_p)$ is full rank even if $\mathbf{X}^t\mathbf{X}$ is not.

Insight into ridge regression

- Compute the SVD of the $n \times p$ input matrix $\mathbf{X}$ then

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^t$$

  where

  - $\mathbf{U}$ is an $n \times p$ orthogonal matrix
  - $\mathbf{V}$ is a $p \times p$ orthogonal matrix
  - $\mathbf{D}$ is a $p \times p$ diagonal matrix with $d_1 \geq d_2 \geq \cdots \geq d_p \geq 0$.

- Can write **least squares** fitted vector as

$$\mathbf{X}\hat{\beta}^{\mathsf{ls}} = \mathbf{X}(\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t y = \mathbf{U}\mathbf{U}^t y$$

  which is the closest approximation to y in the subspace spanned by the columns of $\mathbf{U}$ ($=$ column space of $\mathbf{X}$).

Insight into Ridge Regression

- Can write **ridge regression** fitted vector as

$$\begin{aligned}\mathbf{X}\hat{\beta}^{\text{ridge}} &= \mathbf{X}(\mathbf{X}^t\mathbf{X}+\lambda\mathbf{I}_p)^{-1}\mathbf{X}^t y \\ &= \mathbf{UD}(\mathbf{UD}^2+\lambda\mathbf{I}_p)^{-1}\mathbf{DU}^t y \\ &= \sum_{j=1}^{p} u_j \frac{d_j^2}{d_j^2+\lambda} u_j^t\, y, \quad \text{where } u_j\text{'s are columns of } \mathbf{U}\end{aligned}$$

- As $\lambda \geq 0 \implies d_j^2/(d_j^2+\lambda) \leq 1$
- Ridge regression computes the coordinates of y wrt to the orthonormal basis of the columns of $\mathbf{U}$.
- It then shrinks these coordinates by the factors $d_j^2/(d_j^2+\lambda)$.
- More shrinkage applied to basis vectors with **smaller** d_j^2.

- Can write **ridge regression** fitted vector as

$$\begin{aligned}\mathbf{X}\hat{\beta}^{\text{ridge}} &= \mathbf{X}(\mathbf{X}^t\mathbf{X} + \lambda\mathbf{I}_p)^{-1}\mathbf{X}^t y \\ &= \mathbf{U}\mathbf{D}(\mathbf{U}\mathbf{D}^2 + \lambda\mathbf{I}_p)^{-1}\mathbf{D}\mathbf{U}^t y \\ &= \sum_{j=1}^{p} u_j \frac{d_j^2}{d_j^2 + \lambda} u_j^t\, y, \quad \text{where } u_j\text{'s are columns of } \mathbf{U}\end{aligned}$$

- As $\lambda \geq 0 \implies d_j^2/(d_j^2 + \lambda) \leq 1$

- Ridge regression computes the coordinates of y wrt to the orthonormal basis of the columns of $\mathbf{U}$.

- It then shrinks these coordinates by the factors $d_j^2/(d_j^2 + \lambda)$.

- More shrinkage applied to basis vectors with **smaller** d_j^2.

- Can write **ridge regression** fitted vector as

$$\begin{aligned}
\mathbf{X}\hat{\beta}^{\text{ridge}} &= \mathbf{X}(\mathbf{X}^t\mathbf{X} + \lambda \mathbf{I}_p)^{-1}\mathbf{X}^t y \\
&= \mathbf{U}\mathbf{D}(\mathbf{U}\mathbf{D}^2 + \lambda \mathbf{I}_p)^{-1}\mathbf{D}\mathbf{U}^t y \\
&= \sum_{j=1}^{p} u_j \frac{d_j^2}{d_j^2 + \lambda} u_j^t\, y, \quad \text{where } u_j\text{'s are columns of } \mathbf{U}
\end{aligned}$$

- As $\lambda \geq 0 \implies d_j^2/(d_j^2 + \lambda) \leq 1$
- Ridge regression computes the coordinates of y wrt to the orthonormal basis of the columns of $\mathbf{U}$.
- It then shrinks these coordinates by the factors $d_j^2/(d_j^2 + \lambda)$.
- More shrinkage applied to basis vectors with **smaller** d_j^2.

- Can write **ridge regression** fitted vector as

$$\begin{aligned}\mathbf{X}\hat{\beta}^{\text{ridge}} &= \mathbf{X}(\mathbf{X}^t\mathbf{X} + \lambda\mathbf{I}_p)^{-1}\mathbf{X}^t y \\ &= \mathbf{U}\mathbf{D}(\mathbf{U}\mathbf{D}^2 + \lambda\mathbf{I}_p)^{-1}\mathbf{D}\mathbf{U}^t y \\ &= \sum_{j=1}^{p} u_j \frac{d_j^2}{d_j^2 + \lambda} u_j^t\, y, \quad \text{where } u_j\text{'s are columns of } \mathbf{U}\end{aligned}$$

- As $\lambda \geq 0 \implies d_j^2/(d_j^2 + \lambda) \leq 1$
- Ridge regression computes the coordinates of y wrt to the orthonormal basis of the columns of $\mathbf{U}$.
- It then shrinks these coordinates by the factors $d_j^2/(d_j^2 + \lambda)$.
- More shrinkage applied to basis vectors with **smaller** d_j^2.

- Can write **ridge regression** fitted vector as

$$\begin{aligned}\mathbf{X}\hat{\beta}^{\text{ridge}} &= \mathbf{X}(\mathbf{X}^t\mathbf{X} + \lambda\mathbf{I}_p)^{-1}\mathbf{X}^t y \\ &= \mathbf{UD}(\mathbf{UD}^2 + \lambda\mathbf{I}_p)^{-1}\mathbf{DU}^t y \\ &= \sum_{j=1}^{p} u_j \frac{d_j^2}{d_j^2 + \lambda} u_j^t\, y, \quad \text{where } u_j\text{'s are columns of } \mathbf{U}\end{aligned}$$

- As $\lambda \geq 0 \implies d_j^2/(d_j^2 + \lambda) \leq 1$
- Ridge regression computes the coordinates of y wrt to the orthonormal basis of the columns of $\mathbf{U}$.
- It then shrinks these coordinates by the factors $d_j^2/(d_j^2 + \lambda)$.
- More shrinkage applied to basis vectors with **smaller** d_j^2.

Insight into Ridge Regression: What are the d_j^2's ?

- The sample covariance matrix of the data is given by

$$S = \frac{1}{n}\mathbf{X}^t\mathbf{X}$$

From the SVD of $\mathbf{X}$ we know that

$$\mathbf{X}^t\mathbf{X} = \mathbf{V}\mathbf{D}^2\mathbf{V}^t \quad \longleftarrow \text{eigen-decomposition of } \mathbf{X}^t\mathbf{X}$$

- The eigenvectors v_j - columns of $\mathbf{V}$ are the **principal component directions** of $\mathbf{X}$.

- Project the input of each training example onto the first principal component direction v_1 to get $z_i^{(1)} = v_1^t x_i$. The variance of the $z_i^{(1)}$'s is given by (remember x_i's are centred)

$$\frac{1}{n}\sum_{i=1}^{n}(z_i^{(1)})^2 = \frac{1}{n}\sum_{i=1}^{n} v_1^t x_i x_i^t v_1 = \frac{1}{n} v_1^t X^t X v_1 = \frac{d_1^2}{n}$$

Insight into Ridge Regression: What are the d_j^2's ?

- The sample covariance matrix of the data is given by

$$S = \frac{1}{n}\mathbf{X}^t\mathbf{X}$$

From the SVD of $\mathbf{X}$ we know that

$$\mathbf{X}^t\mathbf{X} = \mathbf{V}\mathbf{D}^2\mathbf{V}^t \quad \longleftarrow \text{eigen-decomposition of } \mathbf{X}^t\mathbf{X}$$

- The eigenvectors v_j - columns of $\mathbf{V}$ are the **principal component directions** of $\mathbf{X}$.

- Project the input of each training example onto the first principal component direction v_1 to get $z_i^{(1)} = v_1^t x_i$. The variance of the $z_i^{(1)}$'s is given by (remember x_i's are centred)

$$\frac{1}{n}\sum_{i=1}^{n}(z_i^{(1)})^2 = \frac{1}{n}\sum_{i=1}^{n} v_1^t x_i x_i^t v_1 = \frac{1}{n} v_1^t X^t X v_1 = \frac{d_1^2}{n}$$

Insight into Ridge Regression: What are the d_j^2's ?

- The sample covariance matrix of the data is given by

$$S = \frac{1}{n}\mathbf{X}^t\mathbf{X}$$

From the SVD of $\mathbf{X}$ we know that

$$\mathbf{X}^t\mathbf{X} = \mathbf{V}\mathbf{D}^2\mathbf{V}^t \quad \longleftarrow \text{eigen-decomposition of } \mathbf{X}^t\mathbf{X}$$

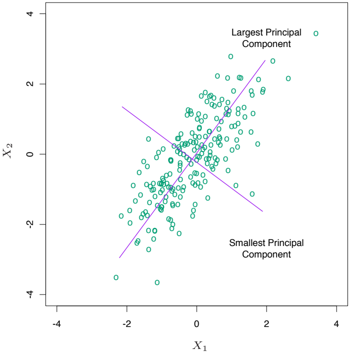
- The eigenvectors v_j - columns of $\mathbf{V}$ are the **principal component directions** of $\mathbf{X}$.

- Project the input of each training example onto the first principal component direction v_1 to get $z_i^{(1)} = v_1^t x_i$. The variance of the $z_i^{(1)}$'s is given by (remember x_i's are centred)

$$\frac{1}{n}\sum_{i=1}^{n}(z_i^{(1)})^2 = \frac{1}{n}\sum_{i=1}^{n} v_1^t x_i x_i^t v_1 = \frac{1}{n} v_1^t X^t X v_1 = \frac{d_1^2}{n}$$

Insight into Ridge Regression: What are the d_j^2's ?

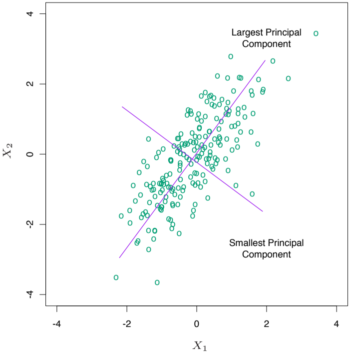
- v_1 represents the direction (of unit length) which the projected points have largest variance.



- Subsequent principal components $z_i^{(j)}$ have maximum variance d_j^2/n subject to v_j being orthogonal to the earlier directions.

Insight into Ridge Regression: What are the d_j^2's ?

- v_1 represents the direction (of unit length) which the projected points have largest variance.



- Subsequent principal components $z_i^{(j)}$ have maximum variance d_j^2/n subject to v_j being orthogonal to the earlier directions.

Insight into Ridge Regression: What are the d_j^2's ?

- The last principal component has minimum variance.
- Hence the small d_j correspond to the directions of the column space of $\mathbf{X}$ having small variance.
- Ridge regression shrinks these directions the most !
- The estimated directions v_j's with small d_j have more uncertainty associated with the estimate. (Using a narrow baseline to estimate a direction). Ridge regression protects against relying on these high variance directions.
- Ridge regression implicitly assumes that the output will vary most in the directions of the high variance of the inputs. A reasonable assumption but not always true.

Insight into Ridge Regression: What are the d_j^2's ?

- The last principal component has minimum variance.
- Hence the small d_j correspond to the directions of the column space of $\mathbf{X}$ having small variance.
- Ridge regression shrinks these directions the most !
- The estimated directions v_j's with small d_j have more uncertainty associated with the estimate. (Using a narrow baseline to estimate a direction). Ridge regression protects against relying on these high variance directions.
- Ridge regression implicitly assumes that the output will vary most in the directions of the high variance of the inputs. A reasonable assumption but not always true.

Insight into Ridge Regression: What are the d_j^2's ?

- The last principal component has minimum variance.
- Hence the small d_j correspond to the directions of the column space of $\mathbf{X}$ having small variance.
- Ridge regression shrinks these directions the most !
- The estimated directions v_j's with small d_j have more uncertainty associated with the estimate. (Using a narrow baseline to estimate a direction). Ridge regression protects against relying on these high variance directions.
- Ridge regression implicitly assumes that the output will vary most in the directions of the high variance of the inputs. A reasonable assumption but not always true.

Insight into Ridge Regression: What are the d_j^2's ?

- The last principal component has minimum variance.
- Hence the small d_j correspond to the directions of the column space of $\mathbf{X}$ having small variance.
- Ridge regression shrinks these directions the most !
- The estimated directions v_j's with small d_j have more uncertainty associated with the estimate. (Using a narrow baseline to estimate a direction). Ridge regression protects against relying on these high variance directions.
- Ridge regression implicitly assumes that the output will vary most in the directions of the high variance of the inputs. A reasonable assumption but not always true.

Insight into Ridge Regression: What are the d_j^2's ?

- The last principal component has minimum variance.
- Hence the small d_j correspond to the directions of the column space of $\mathbf{X}$ having small variance.
- Ridge regression shrinks these directions the most !
- The estimated directions v_j's with small d_j have more uncertainty associated with the estimate. (Using a narrow baseline to estimate a direction). Ridge regression protects against relying on these high variance directions.
- Ridge regression implicitly assumes that the output will vary most in the directions of the high variance of the inputs. A reasonable assumption but not always true.

Insight into Ridge Regression: What are the d_j^2's ?

- The book defines the **effective degrees of freedom** of the ridge regression fit as

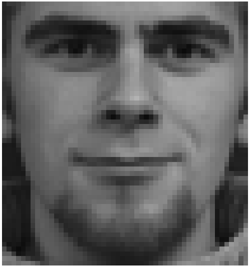
$$\mathrm{df}_{\text{ridge}}(\lambda) = \sum_{j=1}^{p} \frac{d_j^2}{d_j^2 + \lambda}$$

  we will derive this later on in the course.

- But it is interesting as
  - $\mathrm{df}_{\text{ridge}}(\lambda) \to p$ when $\lambda \to 0$ (ordinary least squares) and
  - $\mathrm{df}_{\text{ridge}}(\lambda) \to 0$ when $\lambda \to \infty$

Back to our regression problem

Regression Example: Face Landmark Estimation



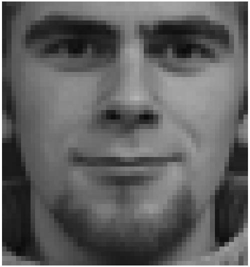
Input

$$\xrightarrow{f_{14},\ g_{14}}$$

Output

- Given a test image want to predict each of its facial landmark points.
- How well can ridge regression do on this problem?

Regression Example: Face Landmark Estimation



$$\xrightarrow{f_{14},\ g_{14}}$$

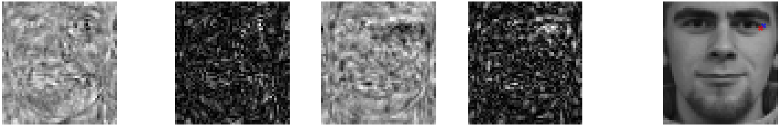
Input **Output**

- Given a test image want to predict each of its facial landmark points.
- How well can ridge regression do on this problem?

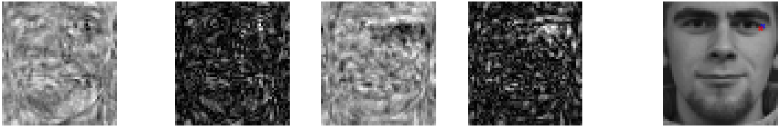
Landmark estimation using ridge regression

$\hat{\beta}_{x,14}$	$\lvert\hat{\beta}_{x,14}\rvert$	$\hat{\beta}_{y,14}$	$\lvert\hat{\beta}_{y,14}\rvert$	Estimated Landmark on novel image

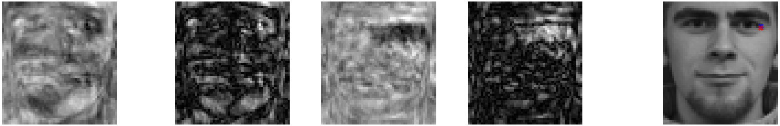
$\lambda = 1$,df ≈ 232



$\lambda = 10^2$,df ≈ 187



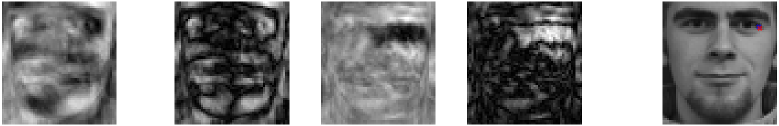
$\lambda = 10^3$,df ≈ 97



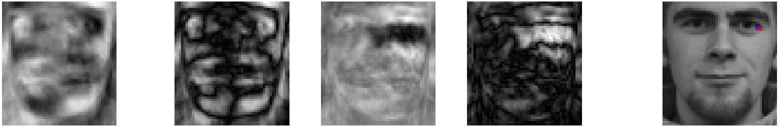
Landmark estimation using ridge regression

$\hat{\beta}_{x,14}$	$\lvert\hat{\beta}_{x,14}\rvert$	$\hat{\beta}_{y,14}$	$\lvert\hat{\beta}_{y,14}\rvert$	Estimated Landmark on novel image

$\lambda = 5000$,df ≈ 44



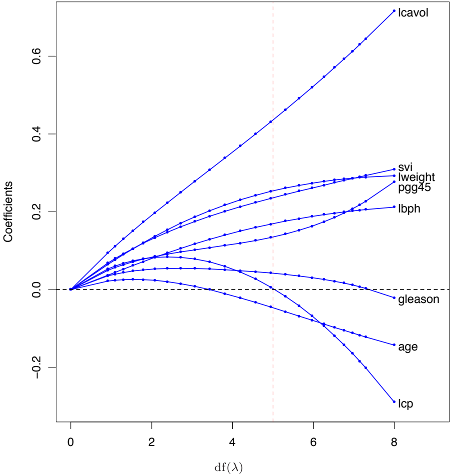
$\lambda = 10^4$,df ≈ 29



Landmark estimation using ridge regression

$\lambda = 1000$,df ≈ 97, Ground truth point, Estimated point

How the coefficients vary with λ



This is an example from the book. Notice how the weights associated with each predictor vary with λ.

The Lasso

- The **lasso** estimate is defined by

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \le t$$

penalty is L_1 instead of L_2 norm

- Equivalent formulation of the lasso problem is

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- The solution is non-linear in y_i's and there is no closed form solution.
- It is convex and is, in fact, a quadratic programming problem.

Shrinkage Method 2: The Lasso

- The **lasso** estimate is defined by

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t$$

penalty is L_1 instead of L_2 norm

- Equivalent formulation of the lasso problem is

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- The solution is non-linear in y_i's and there is no closed form solution.
- It is convex and is, in fact, a quadratic programming problem.

Shrinkage Method 2: The Lasso

- The **lasso** estimate is defined by

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t$$

penalty is L_1 instead of L_2 norm

- Equivalent formulation of the lasso problem is

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- The solution is non-linear in y_i's and there is no closed form solution.
- It is convex and is, in fact, a quadratic programming problem.

Shrinkage Method 2: The Lasso

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \text{ subject to } \sum_{j=1}^{p} |\beta_j| \leq t$$

- Because of the L_1 constraint, making t small will force some of the β_j's to be exactly 0.
- Lasso does some kind of continuous subset selection.
- However the nature of the shrinkage is not so obvious.
- If $t \geq \sum_{j=1}^{p} |\hat{\beta}_j^{\text{ls}}|$ is sufficiently large, then $\hat{\beta}^{\text{lasso}} = \hat{\beta}^{\text{ls}}$.

Shrinkage Method 2: The Lasso

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \text{ subject to } \sum_{j=1}^{p} |\beta_j| \le t$$

- Because of the L_1 constraint, making t small will force some of the β_j's to be exactly 0.
- Lasso does some kind of continuous subset selection.
- However the nature of the shrinkage is not so obvious.
- If $t \ge \sum_{j=1}^{p} |\hat{\beta}_j^{\text{ls}}|$ is sufficiently large, then $\hat{\beta}^{\text{lasso}} = \hat{\beta}^{\text{ls}}$.

Shrinkage Method 2: The Lasso

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t$$

- Because of the L_1 constraint, making t small will force some of the β_j's to be exactly 0.
- Lasso does some kind of continuous subset selection.
- However the nature of the shrinkage is not so obvious.
- If $t \geq \sum_{j=1}^{p} |\hat{\beta}_j^{\text{ls}}|$ is sufficiently large, then $\hat{\beta}^{\text{lasso}} = \hat{\beta}^{\text{ls}}$.

Shrinkage Method 2: The Lasso

$$\hat{\beta}^{\text{lasso}} = \arg\min_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \ \leq t$$

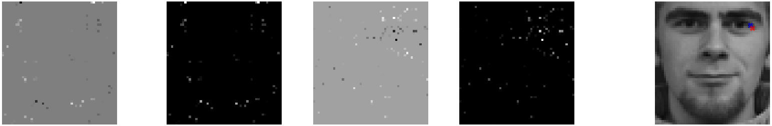
- Because of the L_1 constraint, making t small will force some of the β_j's to be exactly 0.
- Lasso does some kind of continuous subset selection.
- However the nature of the shrinkage is not so obvious.
- If $t \geq \sum_{j=1}^{p} |\hat{\beta}_j^{\text{ls}}|$ is sufficiently large, then $\hat{\beta}^{\text{lasso}} = \hat{\beta}^{\text{ls}}$.

Back to our regression problem

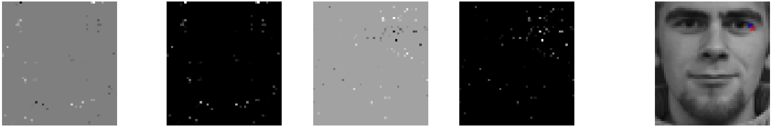
Landmark estimation using lasso

$\hat{\beta}_{x,14}$	$\lvert\hat{\beta}_{x,14}\rvert$	$\hat{\beta}_{y,14}$	$\lvert\hat{\beta}_{y,14}\rvert$	Estimated Landmark on novel image

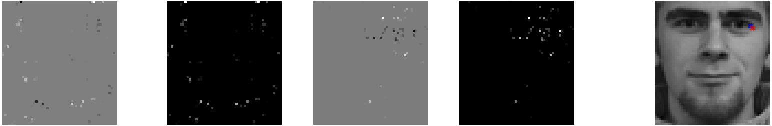
$\lambda = .01$,df $\approx$



$\lambda = .04$,df ≈ 93



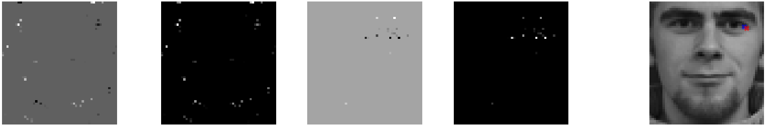
$\lambda = .1$,df ≈ 53



Landmark estimation using lasso

$\hat{\beta}_{x,14}$	$\lvert\hat{\beta}_{x,14}\rvert$	$\hat{\beta}_{y,14}$	$\lvert\hat{\beta}_{y,14}\rvert$	Estimated Landmark on novel image

$\lambda = .3$,df ≈ 21



$\lambda = .5$,df ≈ 7

$\lambda = .7$,df ≈ 2

Landmark estimation using lasso regression

$\lambda = .04$,df ≈ 93, Ground truth point, Estimated point

When $\mathbf{X}$ has orthonormal columns

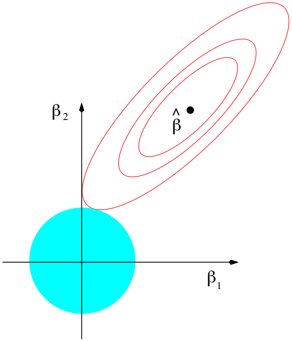
- This implies $d_j = 1$ for $j = 1, \ldots, p$.
- In this case each method applies a simple transformation to $\hat{\beta}_j^{\text{ls}}$:

Estimator	**Formula**
Best subset (size M)	$\hat{\beta}_j^{\text{ls}} \cdot \text{Ind}(\lvert\hat{\beta}_j^{\text{ls}}\rvert \geq \lvert\hat{\beta}_M^{\text{ls}}\rvert)$
Ridge	$\hat{\beta}_j^{\text{ls}}/(1+\lambda)$
Lasso	$\text{sign}(\hat{\beta}_j^{\text{ls}})\,(\lvert\hat{\beta}_j^{\text{ls}}\rvert - \lambda)_+$

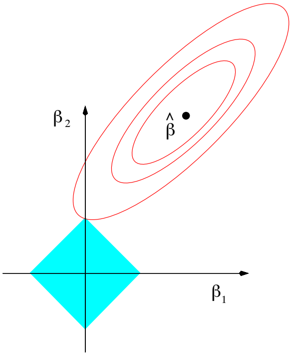
where $\hat{\beta}_M^{\text{ls}}$ is the Mth largest coefficient.

When $\mathbf{X}$ does not have orthogonal columns

- Red elliptical contours show the iso-scores of $\mathrm{RSS}(\beta)$.
- Cyan regions show the feasible regions $\beta_1^2 + \beta_2^2 \leq t^2$ and $|\beta_1| + |\beta_2| \leq t$ resp.



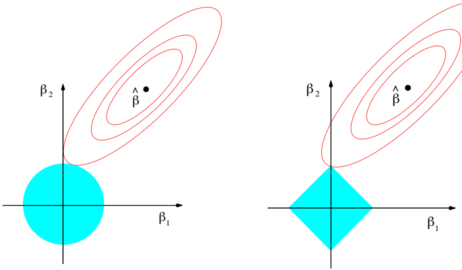
Ridge regression



Lasso

When $\mathbf{X}$ does not have orthogonal columns

- Both methods choose the first point where the elliptical contours hit the constraint region.
- The Lasso region has corners, if then solution occurs at a corner then one $\beta_j = 0$.
- When $p > 2$ the diamond becomes a rhomboid with many corners and flat edges $\implies$ many opportunities for β_j's to be 0.

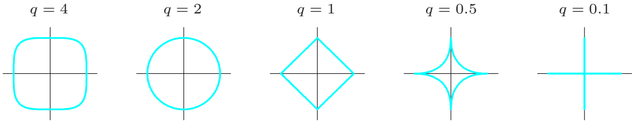


Ridge regression **Lasso**

Generalization of ridge and lasso regression

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|^q \right\}$$

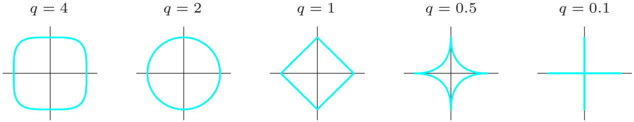
- $q = 0$ - Variable subset selection
- $q = 1$ - Lasso
- $q = 2$ - Ridge regression



Generalization of ridge and lasso regression

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|^q \right\}$$

- Can try other values of q.
- When $q \geq 1$ still have a convex problem.
- When $0 \leq q < 1$ do not have a convex problem.
- When $q \leq 1$ sparse solutions are explicitly encouraged.
- When $q > 1$ cannot set coefficients to zero.



A compromise between the ridge and lasso penalty is the **Elastic net** penalty:

$$\lambda \sum_{j=1}^{p} (\alpha \beta_j^2 + (1-\alpha)|\beta_j|)$$

The elastic-net

- select variables like the lasso **and**
- shrinks together the coefficients of correlated predictors like ridge regression.

Effective degrees of freedom

Definition of the effective degrees of freedom

- Traditionally the number of linearly independent parameters is what is meant by degrees of freedom.
- If we carry out a best subset selection to determine the optimal set of k predictors, then surely we have used more than k dofs.
- A more general definition for the **effective degrees of freedom** of adaptively fitted is

$$\mathrm{df}(\hat{y}) = \frac{1}{\sigma^2}\sum_{i=1}^{n}\mathrm{Cov}(\hat{y}_i, y_i)$$

  where $\mathrm{Cov}(\hat{y}_i, y_i)$ is the estimate of the

- Intuitively the harder we fit to the data, the larger the covariance and hence $\mathrm{df}(\hat{y})$

Definition of the effective degrees of freedom

- Traditionally the number of linearly independent parameters is what is meant by degrees of freedom.
- If we carry out a best subset selection to determine the optimal set of k predictors, then surely we have used more than k dofs.
- A more general definition for the **effective degrees of freedom** of adaptively fitted is

$$\mathrm{df}(\hat{y}) = \frac{1}{\sigma^2} \sum_{i=1}^{n} \mathrm{Cov}(\hat{y}_i, y_i)$$

  where $\mathrm{Cov}(\hat{y}_i, y_i)$ is the estimate of the

- Intuitively the harder we fit to the data, the larger the covariance and hence $\mathrm{df}(\hat{y})$

Definition of the effective degrees of freedom

- Traditionally the number of linearly independent parameters is what is meant by degrees of freedom.
- If we carry out a best subset selection to determine the optimal set of k predictors, then surely we have used more than k dofs.
- A more general definition for the **effective degrees of freedom** of adaptively fitted is

$$\mathrm{df}(\hat{y}) = \frac{1}{\sigma^2} \sum_{i=1}^{n} \mathrm{Cov}(\hat{y}_i, y_i)$$

  where $\mathrm{Cov}(\hat{y}_i, y_i)$ is the estimate of the

- Intuitively the harder we fit to the data, the larger the covariance and hence $\mathrm{df}(\hat{y})$

Definition of the effective degrees of freedom

- Traditionally the number of linearly independent parameters is what is meant by degrees of freedom.
- If we carry out a best subset selection to determine the optimal set of k predictors, then surely we have used more than k dofs.
- A more general definition for the **effective degrees of freedom** of adaptively fitted is

$$\mathrm{df}(\hat{y}) = \frac{1}{\sigma^2} \sum_{i=1}^{n} \mathrm{Cov}(\hat{y}_i, y_i)$$

  where $\mathrm{Cov}(\hat{y}_i, y_i)$ is the estimate of the

- Intuitively the harder we fit to the data, the larger the covariance and hence $\mathrm{df}(\hat{y})$