IEMS 304 Lecture 4: Model and Variable Selection, Shrinkage, and Multicollinearity

Yiping Lu
yiping.lu@northwestern.edu

Industrial Engineering & Management Sciences
Northwestern University

Model Selection

Fitting a Polynomial Using Linear Regression

Consider fitting a polynomial of degree p to data $\{(x_i, y_i)\}$:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_p x^p + \epsilon.$$

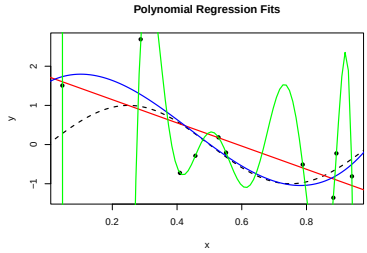
Define new variables: $z_1 = x, \quad z_2 = x^2, \quad \ldots, \quad z_p = x^p$. Then, the model can be written as:

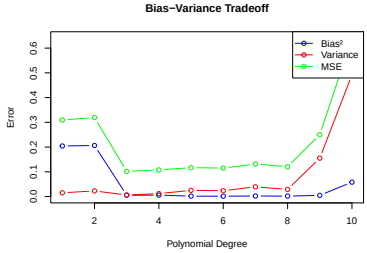
$$y = \beta_0 + \beta_1 z_1 + \beta_2 z_2 + \cdots + \beta_p z_p + \epsilon,$$

which is linear in the parameters $\beta_0, \beta_1, \ldots, \beta_p$.

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^p \\ 1 & x_2 & x_2^2 & \cdots & x_2^p \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^p \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}.$$

Is More Feature Better? (Homework)

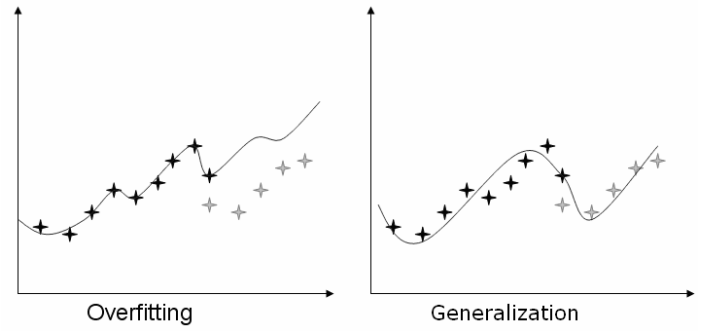




Questions?

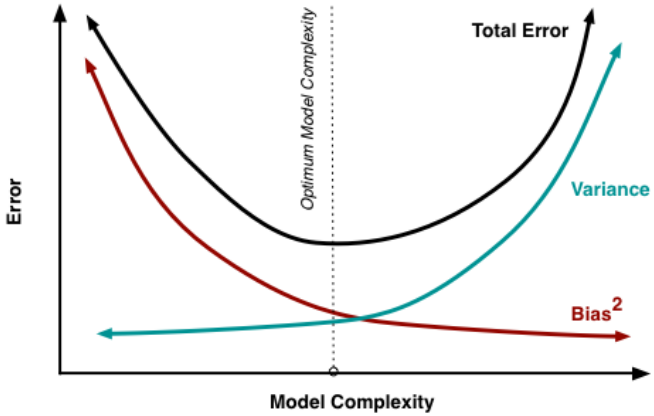
How to Select the Number of Features?

Intuitive Understanding of Model Selection



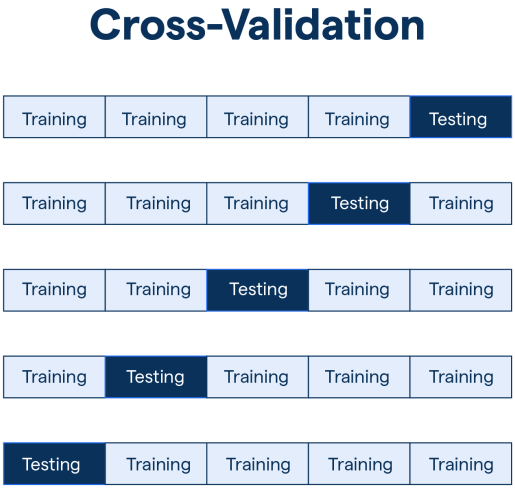
- SSE is small, but prediction error can be large.
- We want to select models that **generalize**.

Bias-Variance Trade-off



First Idea: Cross-Validation

Hold a test s



Recall: Degree of Freedom

Fact 1. $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(y_i' - \hat{y}_i)^2\right] = \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2\right] + \frac{2\sigma^2}{n}\,\mathrm{df}(\hat{y})$.

Fact 2. $\mathrm{df}(\hat{y}^{\mathrm{linreg}}) = p$

How to design a Model Selection algorithm?

Model Selection Algorithms

penalize for larger d and/or larger SSE

Criterion	Large-sample complexity penalization
MSE, r^2_{adj}	d
AIC, C_p	$2d$
BIC	$d \log n$

- $r^2_{\text{adj}} = 1 - \frac{\text{MSE}}{\text{MST}} = 1 - \frac{\text{SSE}}{\text{SST}} \cdot \frac{n-1}{n-(k+1)}$ (larger is better).
- $\text{MSE} = \frac{\text{SSE}}{n-(k+1)}$ (smaller is better).
- Mallow's C_p: $C_p = \frac{\text{SSE}}{\hat{\sigma}^2} + (2d - n)$ (smaller is better),
- AIC: $\frac{1}{n}\left[\frac{\text{SSE}}{\hat{\sigma}^2} + 2d\right]$ (smaller is better)
- BIC: $\frac{1}{n}\left[\frac{\text{SSE}}{\hat{\sigma}^2} + d \cdot \log n\right]$. (smaller is better)

C_p can be viewed as a special case of AIC for linear regression. AIC and BIC (for both, smaller is better) are much more general than C_p and apply to many nonlinear models fit via maximum likelihood estimation (MLE).

Discussion

- When comparing models with the same d, using C_p, r^2_{adj}, r^2, MSE, AIC, or BIC are all equivalent to selecting the model with the lowest training SSE.
- When comparing models with different d, using simply SSE for model selection is usually not a good idea.
- Using C_p, r^2_{adj}, r^2, MSE, AIC, or BIC may lead to different model selection
- For large data sets, CV often gives smaller selected models than any of the analytical criteria.

Analytical Criteria v.s. Cross-Validation

- Cross-validation (CV) can be used to evaluate and compare virtually any set of models.
  - CV applies to any type of model (linear, nonlinear, trees, neural networks, etc)
  - CV applies equally well to classification and regression (but for classification you would use a different error measure than SSE)
  - CV is generally the most reliable, because it involves no assumptions (analytical criteria like C_p, AIC, BIC involve assumptions, such as no influential observations or outliers, large sample sizes, etc)
- CV is too computationally expensive for the automated variable selection methods. For these, we need the analytical criteria. But we can always use CV to assess and compare a few final candidate best models.

Stepwise and Subsets Regression

Variable Selection

- Given a possibly large set of predictor variables $\{x_1, x_2, \ldots, x_k\}$, how to decide which ones belong in the model?
  - Including more predictors than needed is bad for explanatory, as well as predictive, purposes.
  - Could consider doing partial F-tests comparing all possible combinations of models having subsets of the k predictors, but this is not computationally feasible, and we would like a more automated method.
  - Could consider fitting one model with all k predictors and then looking at their t-test P-values (why is this a bad approach?)
- Two common automated variable selection methods are
  - Stepwise regression (good, and computationally feasible);
  - Best subsets regression (best, only feasible for $k < 50$ or so).

Forward Stepwise Regression

- Basic idea is to start with no predictors in the model and build the model iteratively (in steps), one predictor at a time. On each step you:
  - Find which one of the remaining individual predictors would most reduce the SSE if it were added to the model.
  - Use some criterion like AIC to decide whether the model is better with or without that one predictor.
  - If the criterion says to add that one predictor, you add it and go to the next step; otherwise, you terminate the algorithm and take the best model to be the current one.
- The original criterion for deciding whether the model is better with or without the additional predictor was a partial F-test, and this is still used in many software.
- AIC or Mallows' C_p is usually considered preferable now.

A Toy Example of 8 Variables

- Suppose we are at the third iteration to add variables.
- In the first iteration, we added predictor x_2 and at the second iteration we added predictor x_5.
- The current model contains $\{x_2, x_5\}$ and we test the following six combinations:

$$\{x_2, x_5, x_1\} \quad \{x_2, x_5, x_3\} \quad \{x_2, x_5, x_4\}$$
$$\{x_2, x_5, x_6\} \quad \{x_2, x_5, x_7\} \quad \{x_2, x_5, x_8\}$$

- Suppose $\{x_2, x_5, x_1\}$ has the smallest SSE. We denote it as SSE_3. Let SSE_2 denote the SSE for the model $\{x_2, x_5\}$.
- We calculate $\mathrm{AIC}_2 = n\log(\mathrm{SSE}_2) + 2 \times 3$ and $\mathrm{AIC}_3 = n\log(\mathrm{SSE}_3) + 2 \times 4$.
- If $\mathrm{AIC}_3 < \mathrm{AIC}_2$, we add x_1 to the model and proceed to the fourth iteration. Otherwise, we terminate and take $\{x_2, x_5\}$ as the final model.

Forward vs. Backward vs. Forward/Backward

- **Forward Stepwise**: Start with no predictors in the model and add them one-at-a-time.
- **Backward Stepwise**: Start with all k predictors in the model and remove them one-at-a-time. At each step, the removed predictor is the one that least increases the SSE after its removal. Stop removing according to the same AIC or F-test criteria.
- **Forward/Backward Stepwise (forward version)**: Start with no predictors in the model and add them one-at-a-time. However, at each step, you can consider removing one or more of the predictors that were added at a previous step. Whether to add, remove, or stop is determined according to the same AIC or F-test criteria.

Example: Stepwise Regression

- `pred_weight.txt` contains data to predict person's weight. We demonstrate the forward/backward stepwise regression.
- The initial model is a constant model, i.e., $\text{weight} \sim 1$.
- We add predictors one-by-one in each iteration. Meanwhile, in each iteration, we check if any previously added predictors can be removed.

```
step(object, scope, scale = 0,
     direction = c("both", "backward", "forward"),
     trace = 1, keep = NULL, steps = 1000, k = 2, ...)
```

Example: The First Iteration

```
Start:  AIC=205.9
weight ~ 1

             Df Sum of Sq   RSS    AIC
+ gender      1   15232.5 11615 182.76
+ height      1    8968.4 17879 195.71
+ age         1    4034.2 22813 203.02
<none>                    26847 205.90
+ digit       1    1260.7 25587 206.46
+ meat        1     868.7 25979 206.91
+ NL          1     313.6 26534 207.55
+ cell_phone  1     244.6 26603 207.63
+ fruit_veg   1     166.3 26681 207.72
```

- Which added predictor achieves the lowest SSE?
- Shall we add the predictor identified above to the model?

Example: The Second Iteration

```
Step:  AIC=182.77
weight ~ gender

             Df Sum of Sq   RSS    AIC
+ age         1    1223.6 10391 181.43
+ height      1    1088.7 10526 181.81
<none>                    11615 182.76
+ NL          1     313.6 11301 183.94
+ meat        1      64.7 11550 184.60
+ fruit_veg   1       4.9 11610 184.75
+ cell_phone  1       3.8 11611 184.76
+ digit       1       0.4 11614 184.76
- gender      1   15232.5 26847 205.90
```

- Which added predictor achieves the lowest SSE?
- Shall we add the predictor identified above to the model?

Example: The Third and Fourth Iteration

```
Step:  AIC=181.43
weight ~ gender + age

             Df Sum of Sq     RSS    AIC
+ height      1     750.4  9640.8 181.18
<none>                    10391.3 181.43
+ NL          1     313.6 10077.6 182.51
- age         1    1223.6 11614.8 182.76
+ digit       1      50.1 10341.1 183.28
+ meat        1      36.2 10355.0 183.32
+ fruit_veg   1      34.8 10356.4 183.32
+ cell_phone  1       1.9 10389.4 183.42
- gender      1   12421.9 22813.2 203.02
```

```
Step:  AIC=181.18
weight ~ gender + age + height

             Df Sum of Sq     RSS    AIC
<none>                     9640.8 181.18
- height      1     750.4 10391.3 181.43
- age         1     885.3 10526.1 181.81
+ digit       1     404.7  9236.2 181.89
+ NL          1     200.1  9440.7 182.55
+ meat        1      33.1  9607.7 183.07
+ fruit_veg   1      26.4  9614.5 183.09
+ cell_phone  1       3.5  9637.3 183.17
- gender      1    6685.4 16326.2 194.98
```

- Shall we continue the process or terminate?

Questions and Discussions

- Stepwise regression is "fooled" by influential observations (just like other tests of statistical significance of the coefficients are fooled), so this must be taken into account.
- When you have many predictors and suspect that only a few may be important, forward stepwise is preferable to backwards.
- When you suspect that most predictors may be important, backward stepwise may be preferable.
- Suppose you have 50 rows of data, 75 predictor variables, and you are not sure how many of the 75 are important. Would backwards or forwards stepwise be a better choice in this case?

Best Subsets Regression

- Basic Idea: For $p = 1, 2, \ldots, k$, find the best (or best 2 or 3) models that contain exactly p predictors, a subset of $\{x_1, x_2, \ldots, x_k\}$.
- You can then choose the overall best model from among the best of each size.
- How to quantify which models are "better"?
  - For comparing models having the same p, this is easy: better = lower SSE.
  - For comparing models having different p, you can use your favorite model selection criterion (C_p, AIC, CV, etc.).

Example: Best Subsets Regression

- We use `pred_weight.txt` data again. The leaps() function is useful for best subsets regression.

	size	Cp	height	gender	meat	fruit_veg	age	cell_phone	digit	NL
X1	2	1.727439	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
X1.1	2	16.681476	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
X2	3	0.806466	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE
X2.1	3	1.128335	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
X3	4	1.014981	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE
X3.1	4	2.057747	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
X4	5	2.048960	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE
X4.1	5	2.537256	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
X5	6	3.098190	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE
X5.1	6	3.899458	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE
X6	7	5.050088	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE
X6.1	7	5.090002	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE
X7	8	7.008034	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE
X7.1	8	7.046338	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
X8	9	9.000000	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

Questions and Discussions

- Best subsets with r^2_{adj} as the criterion (use method = "adjr2") would give a 5-predictor model with $\{\text{gender}, \text{age}, \text{height}, \text{digit}, \text{NL}\}$ as the best model, which is clearly too many predictors. In contrast, using C_p as the criterion gives the 2-predictor model $\{\text{gender}, \text{age}\}$ as the best model.
- The top three models in order of C_p are $\{\text{gender}, \text{age}\}$, $\{\text{gender}, \text{age}, \text{height}\}$, and $\{\text{gender}, \text{height}\}$.
- These three models have similar C_p. What follow-up analyses would you do to decide which is the best model?

Follow-up Analysis After Best Subsets

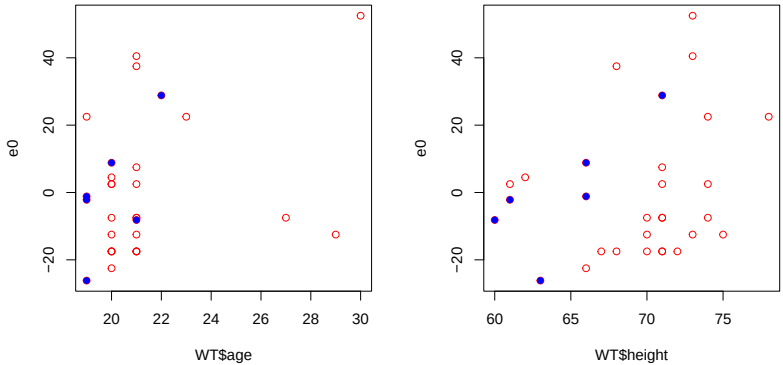
- Variable gender is in all of the top models. We explore whether height or age is the better predictor to include.
- We use PRESS to evaluate those models again.

Model	C_p	PRESS
{gender, age}	0.8	14858
{gender, height, age}	1.0	14582
{gender, height}	1.1	12720
{gender}	1.7	13232

- According to PRESS, {height, gender} is the best model.

Why age Loses the Game?

- We try to fit a simple linear model $\text{weight} \sim \text{gender}$ and do some residual plots.
- We distinguish the residuals according to gender. For male, the residual is represented in blue and for female, the residual is represented in red.



Stepwise v.s. Best Subsets Regression

- Computational (major advantage for stepwise):
  - Stepwise is very fast computationally and can handle virtually any number of predictors, even with large data sets.
  - Best subsets is very slow even with the computational tricks. It cannot handle more than k $>$ 50 predictors, or so.
- Optimality of selected model (minor advantage for best subsets)
  - Stepwise is a greedy optimization algorithm that does not necessarily find best model of each size (for fixed size, best means lowest SSE), although it usually does a pretty good job.
  - Best subsets is guaranteed to find the best model of each size.
- Flexibility (major advantage for stepwise):
  - Versions of stepwise are available for other models, like logistic regression. Best subsets is restricted to linear regression models, because of the computational challenges.

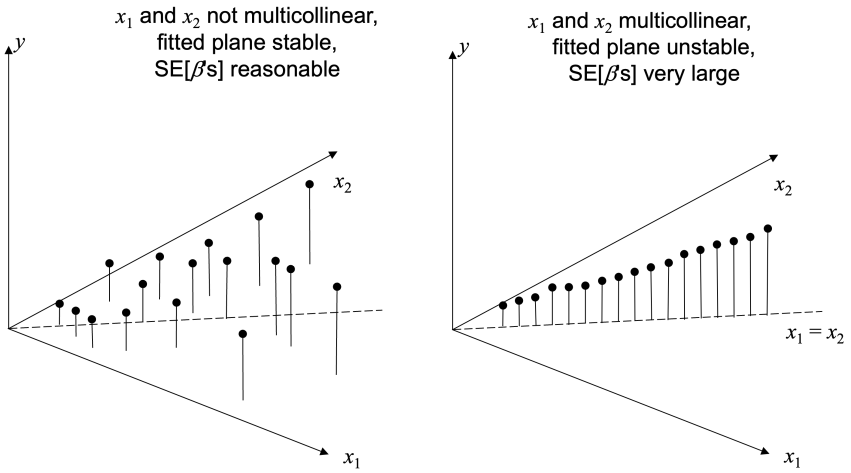
Multicollinearity

Multicollinearity

- Multicollinearity means that some of the predictors (or linear combinations of them) are **highly correlated** with each other.
- We have already seen how multicollinearity causes problems in regression (e.g., misleading t-tests, estimated coefficients that have the wrong sign). It also compounds problems associated with leverage and influence (easier to have high-leverage observations when multicollinearity is present) and causes numerical problems.
- Multicollinearity is closely connected to variable selection:
  - It makes variable selection ambiguous;
  - Variable selection is one "solution" to multicollinearity, since it tends to omit predictors that are correlated with included ones.

Illustration of Multicollinearity

- We fit a model $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$.



Questions and Discussions

- In the right figure on the previous slide, the multicollinearity between x_1 and x_2 makes it nearly impossible to distinguish between their effects. This means we cannot distinguish between β_1 and β_2, which translates to poor estimation and large standard errors.
- Why is the situation depicted in the right figure more likely to be subject to influential observations?
- If you are only interested in predicting the response (i.e., you are not interested in distinguishing the effects of x_1 and x_2), AND you will not be extrapolating/predicting the response at x values that fall outside the relationship seen in the training data (i.e., off the $x_1 = x_2$ line in the right figure), then multicollinearity may not be a problem.

Mathematical Reason Why Multicollinearity Causes Problems

- Recall that we can represent data as a matrix X:

$$\mathrm{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{15} & \dots & x_{1k} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{n5} & \dots & x_{nk} \end{bmatrix}.$$

- Suppose the second and the fifth predictor variables are highly linearly dependent.
- This says matrix X is "almost not full column rank".
- When we solve the linear equations for the coefficients, i.e., $(\mathrm{X}^\top \mathrm{X})\hat{\boldsymbol{\beta}} = \mathrm{X}^\top \mathrm{Y}$, the solution is underdetermined—$\mathrm{X}^\top \mathrm{X}$ is almost singular.

Detecting Multicollinearity

- Inspect matrix scatter plots of predictors (BEWARE: can miss multicollinearity if $k > 2$).
- Inspect correlation matrices of predictors (BEWARE for same reason).
- Variance Inflation Factors (VIFs) (the best way to detect multicollinearity).

Pairwise Multicollinearity

- If you see high correlation (among predictors) in a matrix scatterplot, then multicollinearity is present. However, if you do not see it, it may **still be present**.
- Inspecting correlation matrices is subject to the same pitfall.
- Side note: It is common to standardize the predictors before fitting a model (i.e., standardize each "column" to have zero mean and unit variance)

$$\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij} \quad \text{sample average of } j\text{-th predictor,}$$

$$s_{x_j} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2} \quad \text{sample std of } j\text{-th predictor,}$$

$$x^*_{ij} = \frac{x_{ij} - \bar{x}_j}{s_{x_j}} \quad \text{standardized } j\text{-th predictor.}$$

Correlation Matrix

- We define $r_{x_j x_l} = \frac{1}{n-1} \sum_{i=1}^{n} x_{ij}^* x_{il}^*$ as the sample correlation coefficient between x_j and x_l.
- Correlation matrix is to collect all the correlation coefficients between pairwise predictor, i.e.,

$$\mathrm{R} = \begin{bmatrix} 1 & r_{x_1 x_2} & \dots & r_{x_1 x_k} \\ r_{x_2 x_1} & 1 & \dots & r_{x_2 x_k} \\ \vdots & \vdots & \ddots & \vdots \\ r_{x_k x_1} & r_{x_k x_2} & \dots & 1 \end{bmatrix}.$$

- Interpretation of correlation coefficients:
  - $-1 \leq r_{x_j x_l} \leq 1$ always;
  - $r_{x_j x_l} = \pm 1$ perfectly linearly related;
  - $r_{x_j x_l} = 0$ no (linear) relation.

Example: Correlation Matrix

- In `gas_mileage.csv` data, we calculate the correlation matrix. Part of the matrix is shown below.

	Displacement	Hpower	Torque	Comp_ratio	Rear_axle_ratio
Displacement	1.000	0.945	0.989	-0.330	-0.632
Hpower	0.945	1.000	0.964	-0.292	-0.517
Torque	0.989	0.964	1.000	-0.326	-0.673
Comp_ratio	-0.330	-0.292	-0.326	1.000	0.374
Rear_axle_ratio	-0.632	-0.517	-0.673	0.374	1.000
Carb_barrels	0.659	0.772	0.653	-0.049	-0.205
No._speeds	-0.781	-0.643	-0.746	0.494	0.843
Length	0.855	0.797	0.864	-0.258	-0.548
Width	0.801	0.718	0.788	-0.319	-0.434
Weight	0.946	0.883	0.943	-0.277	-0.542
Trans._type	0.835	0.727	0.801	-0.368	-0.703

Example: (Lurking) Multicollinearity

- `barstock.csv` contains 30 observed cases of 5 variables. Each row is the weight, volume, height, width, and length of a roughly cube-shaped piece of stock metal.
- We can find the correlation matrix as follows.

	volume	height	width	length
volume	1.000	0.369	0.548	0.738
height	0.369	1.000	-0.361	0.054
width	0.548	-0.361	1.000	0.182
length	0.738	0.054	0.182	1.000

Shrinkage

James-Stein Estimator

Used for estimating the mean vector $\theta = (\theta_1, \ldots, \theta_p)$ of a multivariate normal distribution given an observation $X \sim N(\theta, \sigma^2 I_p)$

- **Maximum likelihood estimator:** The sample mean X
- **James-Stein Estimator:** Instead of using the MLE directly, shrink it towards zero (Why?) to reduce the mean squared error (MSE)

$$\hat{\theta}_{JS} = \left(1 - \frac{(p-2)\sigma^2}{\|X\|^2}\right) X, \quad \text{for } p \geq 3$$

Notable Result: The James-Stein estimator dominates the MLE under squared error loss when $p \geq 3$

Example: Risk Comparison for $\theta = 0,\ p = 3$

MLE Estimator: $R(0, \hat{\theta}_{\text{MLE}}) = E\|X - 0\|^2 = E\|X\|^2 = 3.$

James-Stein Estimator: $\hat{\theta}_{\text{JS}} = \left(1 - \frac{1}{\|X\|^2}\right) X.$

Risk Calculation: $R(0, \hat{\theta}_{\text{JS}}) = \left(1 - \frac{1}{\|X\|^2}\right)^2 \|X\|^2 = \|X\|^2 - 2 + \frac{1}{\|X\|^2}.$

- Since $\|X\|^2 \sim \chi^2_3$:
  - $E[\|X\|^2] = 3.$
  - For $\nu > 2$, $E\left[\frac{1}{\|X\|^2}\right] = \frac{1}{\nu - 2}$; hence for $\nu = 3$, $E\left[\frac{1}{\|X\|^2}\right] = 1.$

$$R(0, \hat{\theta}_{\text{JS}}) = 3 - 2 + 1 = 2.$$

James–Stein as Penalized Least Squares

Penalized Objective Function

Consider the objective function

$$J(\theta) = \|X - \theta\|^2 + \lambda \|\theta\|^2,$$

where $X \sim N(\theta, I_p)$ and λ is a penalty parameter.

The minimizer of $J(\theta)$ is found by setting the derivative with respect to θ to zero:

$$\frac{\partial J(\theta)}{\partial \theta} = -2(X - \theta) + 2\lambda\theta = 0.$$

This yields $\hat{\theta} = \frac{1}{1+\lambda} X$. We take $\lambda = \frac{p-2}{\|X\|^2 - (p-2)}$.

Regularized Linear Regression

- Basic idea: When fitting a regression model, instead of minimizing the SSE, pick a small $\lambda > 0$ and minimize

$$\sum_{i=1}^{n}(y_i - \hat{\boldsymbol{\beta}}^\top \mathsf{x}_i)^2 + \lambda \sum_{j=0}^{k} \beta_j^2.$$

  Note that we have added a 1 in each data point x_i.

- Because the objective function is still quadratic in the $\hat{\boldsymbol{\beta}}$, there is a closed form solution:

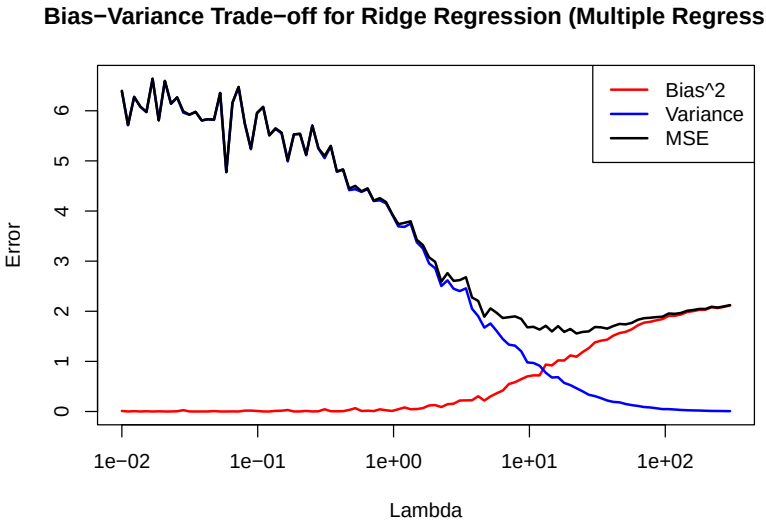
$$\hat{\boldsymbol{\beta}}_{\mathrm{ridge}} = (\mathsf{X}^\top \mathsf{X} + \lambda \mathsf{I})^{-1} \mathsf{X}^\top \mathsf{Y}.$$

- This is called "shrinkage" because $\left\|\hat{\boldsymbol{\beta}}_{\mathrm{ridge}}\right\|_2 \leq \left\|\hat{\boldsymbol{\beta}}\right\|_2$.

Implementing Ridge Regression

- Important: Standardize all predictors first.
- Choose a large initial λ (e.g., $\lambda = n$).
- Fit the ridge regression model.
- Reduce λ (i.e., reset $\lambda \to \lambda/1.5$) and go to the previous step. Repeat until $\lambda \approx 0$.
- Choose the best value of λ by either:
  - inspecting a plot of $\hat{\boldsymbol{\beta}}_{\text{ridge}}$ versus λ and choosing the smallest λ after which $\hat{\boldsymbol{\beta}}_{\text{ridge}}$ stabilizes.
  - C_p with the "model complexity" d replaced by the equivalent number of fitted parameters $\text{trace}(\mathbf{X}[\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}]^{-1}\mathbf{X}^\top)$.
  - Generalized cross-validation (GCV), similar to AIC and C_p.
  - Whatever criterion your software has (there are a few other analytical criteria).
  - As always, cross-validation (see Lab 2) can be used.

Bias-Variance Trade-off

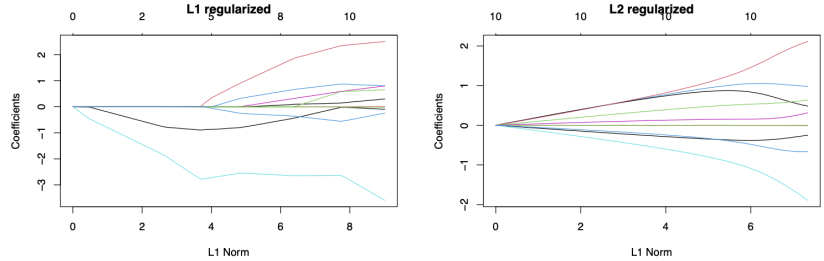


Selecting λ

LASSO

When fitting a regression model, instead of minimizing the SSE, pick a small $\lambda > 0$ and minimize

$$\sum_{i=1}^{n}(y_i - \hat{\boldsymbol{\beta}}^\top \mathrm{x}_i)^2 + \lambda \sum_{j=0}^{k} |\beta_j|.$$



Why?

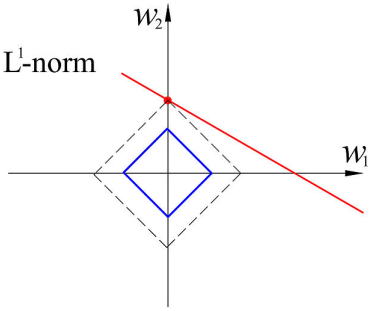
Weight Decay

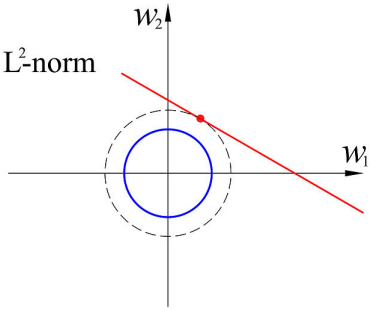
Try to run gradient descent for $F(\beta) + \lambda \underbrace{\sum_{i=1}^{d} \beta_i^2}_{:=\|\beta\|_2^2}$

Gradient Descent gives $\beta_i = (1 - 2\alpha\lambda)\beta_{i-1} - \alpha\nabla F(\beta_{i-1})$

Try to run gradient descent for $F(\beta) + \lambda \underbrace{\sum_{i=1}^{d} |\beta_i|}_{\|\beta\|_1}$

L2 VS L1





New View of Gradient Descent

$\beta_i = \beta_{i-1} - \alpha \nabla F(\beta_{i-1})$ is the solution to

$$\arg\min_{\beta} \underbrace{F(\beta_{i-1}) + \nabla F(\beta_{i-1})(\beta - \beta_{i-1}) + \frac{\alpha}{2}\|\beta - \beta_{i-1}\|_2^2}_{\text{approximation to } F(\beta)}$$

Let's go back to LASSO objective $\underbrace{F(\beta)}_{\text{smooth}} + \underbrace{\lambda \sum_{i=1}^{d} |\beta_i|}_{\text{non-smooth}}$, thus we can update β_i as

$$\arg\min_{\beta} \underbrace{F(\beta_{i-1}) + \nabla F(\beta_{i-1})(\beta - \beta_{i-1}) + \frac{\alpha}{2}\|\beta - \beta_{i-1}\|_2^2}_{\text{approximation to } F(\beta)} + \lambda\|\beta\|_1$$

- Call "Proximal Gradient Descent"

❒ Closed form! (Homework)

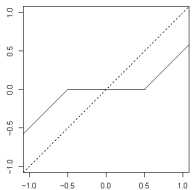
Iterative Shrinkage Thresholding Algorithm (ISTA)

$$\arg\min_\beta \underbrace{F(\beta_{i-1}) + \nabla F(\beta_{i-1})(\beta - \beta_{i-1}) + \frac{\alpha}{2}\|\beta - \beta_{i-1}\|_2^2}_{\text{approximation to } F(\beta)} + \lambda\|\beta\|_1$$

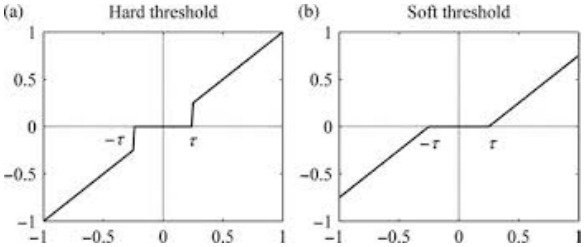
The minimization leads to the update:

$$\beta_i = S_{\lambda/\alpha}\Big(\beta_{i-1} - \frac{1}{\alpha}\nabla F(\beta_{i-1})\Big)$$

where $S_\theta(z) = \text{sign}(z)\max(|z| - \theta, 0)$ is the soft-thresholding operator.



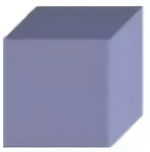
Soft/Hard Thresholding



Hard Thresholding is the proximal algorithm for $F(\beta) + \lambda\|\beta\|_0$ where $\|\beta\|_0$ is the number of 0 coeficents in β_i.

L_p-norm Ball

Why L_1 is so special?



$p = \infty$

$p = 2$

$p = 1$

$0 < p < 1$

$p = 0$

Take `IEMS 351: Optimization Methods in Data Science`