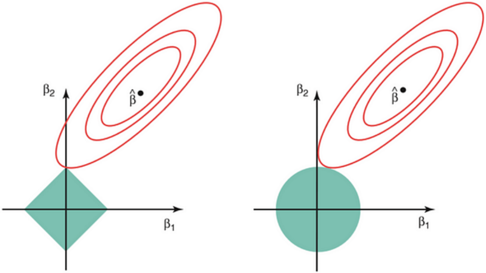
Regularization

PSY-GS 8875 Behavioral Data Science



Overview

Overview: Week 4

Readings

- ESL Chapters: 3.4, 3.4.1, 3.4.2, and 4.4.4
- HML: Chapter 6

Optional

- Jacobucci et al. - 2016
- Seeboth and Mõttus - 2018

- Ridge (ℓ_2-norm) regression
- LASSO (ℓ_1-norm) regression
- Activity: predicting life outcomes with personality

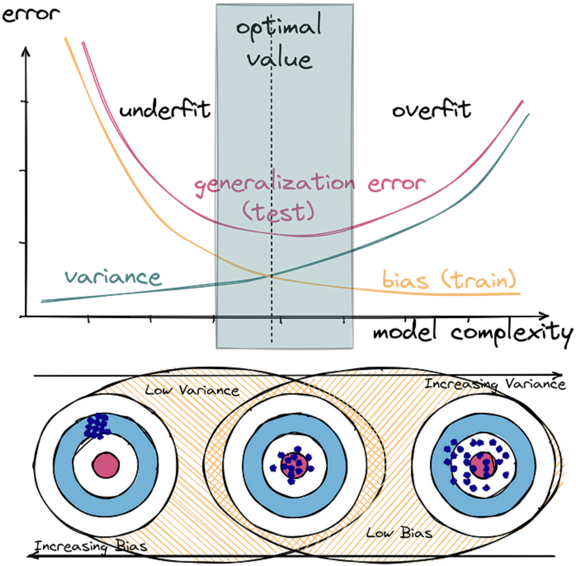
Regularization

Regularization

generalizability: extent to which a model can make predictions beyond the data it was fit

- Most models aim to fit the data the best it can (e.g, OLS)
- The data are the data – the data we have are the best we know of what represents the population

Regularization



Regularization

Premise

- Most models aim to fit the data the best it can (e.g, OLS)
- If we know that our model is overfitting the data we have (high variance, low bias), then we might want to introduce some bias to reduce the overfitting
- Said differently, we might want to purposefully underfit our model to the data we have with the goal to better generalize to other samples

Regularization

Methods

- **Ridge** (ℓ_2-norm) regression
- **LASSO** (ℓ_1-norm) regression
- Elastic net (mix of ridge and LASSO)

Ridge Regression

Ridge Regression

Ridge Regression

- Shrink regression parameters toward zero based on some penalty
- Especially if there are fewer observations than there are variables ($n << p$)
- Multicollinearity can also be reduced by shrinking coefficients (recall that multicollinearity can inflate estimated coefficients)

Ridge Regression

Recall

$$\widehat{\beta} = (\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$$

and

$$\widehat{\beta} = \arg\min_{\widehat{\beta}} \sum (\widehat{\mathbf{y}} - \mathbf{y})^2$$

Ridge Regression

$$\widehat{\beta}_{ridge} = (\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda \mathbf{I})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$$

and

$$\widehat{\beta}_{ridge} = \arg\min_{\widehat{\beta}} \sum (\widehat{\mathbf{y}} - \mathbf{y})^2 + \lambda \sum \beta^2$$

What did we add to these equations?

Ridge Regression

$$\widehat{\beta}_{ridge} = (\mathbf{X^T X} + \lambda \mathbf{I})^{-1} \mathbf{X^T y}$$

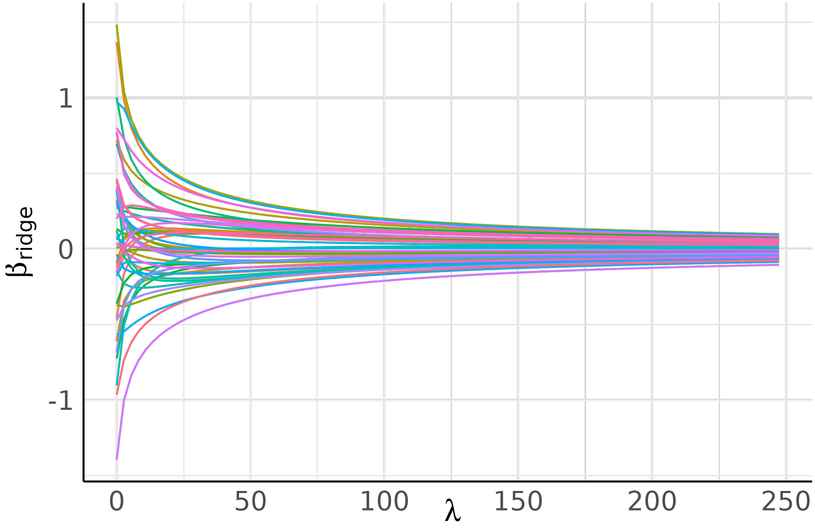
and

$$\widehat{\beta}_{ridge} = \arg\min_{\widehat{\beta}} \sum (\widehat{\mathbf{y}} - \mathbf{y})^2 + \lambda \sum \beta^2$$

What did we add to these equations?

What is $\lambda = 0$?

Ridge Regression



Ridge Regression

R Example

Dataset

- 50 items from the Big Five IPIP inventory [source]
- 428 people (subsampled from the original $N = 9,790$)
- 313 people in a *different* subsample to *test*
- Outcome: total score on well-being measured by the Warwick-Edinburgh Mental Well-Being Scale
- Published analyses using these data: Seeboth and Mõttus - 2018

Head over to the regularization.R script

Optimal λ

Ridge Regression | Optimal λ

- Choosing λ shouldn't be arbitrary
- What might be some ways to select λ? (what criterion/methods have we learned about?)

- Choosing λ shouldn't be arbitrary
- What might be some ways to select λ? (what criterion/methods have we learned about?)
- *k*-folds cross-validation to minimize mean squared error or accuracy is common

Template in R

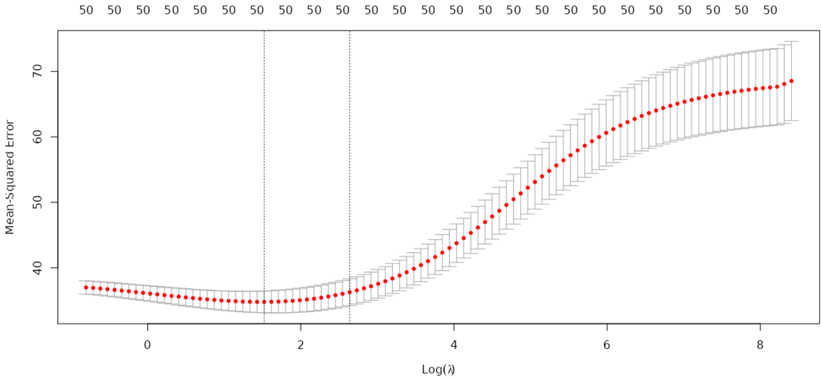
```r
# Set seed for reproducibility
set.seed(42) # don't forget!!

# Perform cross-validation
ridge_cv <- cv.glmnet(
  x = X, # predictors
  y = Y, # outcome
  alpha = 0, # 0 = ridge; 1 = lasso
  nfolds = 10 # number of folds
)

# Print/plot summary
ridge_cv; plot(ridge_cv)
```

By default, glmnet standardizes your variables

Perform Cross-validation to Obtain λ

- Use the `ncds_sample.RData` dataset (don't forget to only keep `complete.cases`)
  - Should have $n = 383$
- Set a seed
- Using the following for your predictors and outcome:
  - Predictors: `ncds_sample[,2:51]`
  - Outcome: `ncds_sample[,"wem_well_being"]`
- Perform cross-validation ridge regularization using 5-folds
- `print` and `plot` the output: What is the `min` λ?

Ridge Regression | Optimal λ




```
   Lambda Index Measure    SE Nonzero
min   4.58    75   34.79 1.660      50
1se  13.99    63   36.28 2.043      50
```

Difference in Coefficients

```r
# Obtain coefficients of best ridge
ridge_best <- glmnet(
  x = X, y = Y, family = "gaussian",
  alpha = 0, lambda = ridge_cv$lambda.min
)

# Standard linear model
standard_lm <- lm(wem_well_being ~ ., data = ncds_sample)

# Compute difference between standard and ridge coefficients
mean(abs(coef(ridge_best)[-1] - coef(standard_lm)[-1]))
```

```
[1] 0.2050615
```

```r
range(abs(coef(ridge_best)[-1] - coef(standard_lm)[-1]))
```

```
[1] 0.0002459013 0.5770982055
```

```r
# Remember the scaling factor with {glmnet} = sd(Y) / length(Y)
```

Generalizability?

Predict New Sample

- Load in the `ncds_test.RData` dataset (don't forget to only keep `complete.cases`)
  - Should have $n = 288$
- Get predictions from standard and ridge models
  - Standard: `predict(standard_lm, newdata = ncds_test)`
  - Ridge: `predict(ridge_best, newx = X_test)`
- Compute RMSE for both standard and ridge model
- Which model generalized better?

New Sample:

```
Standard RMSE: 0.3379202
```

```
Ridge RMSE: 0.3326264
```

Did we generalize better?

Ridge Regression | Generalizability?

New Sample:

`Standard RMSE: 0.3379202`

`Ridge RMSE: 0.3326264`

Did we generalize better?

What about the original sample?

`Standard RMSE: 0.00000000000003376473`

`Ridge RMSE: 0.000000000000005325026`

How about R^2?

	Train	Test	# of Predictors
Standard	0.609	0.447	50
Ridge	0.578	0.490	50

Do you prefer standard or ridge?

LASSO Regression

Least Absolute and Shrinkage Selection Operator (LASSO) Regression

LASSO Regression

- Shrink regression parameters toward zero based on some penalty
- Especially if there are fewer observations than there are variables ($n << p$)
- Multicollinearity can also be reduced by shrinking coefficients (recall that multicollinearity can inflate estimated coefficients)
- Set a “soft-threshold” to shrink small parameter estimates to zero

LASSO Regression

Recall

$$\widehat{\beta} = (\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$$

and

$$\widehat{\beta} = \arg\min_{\widehat{\beta}} \sum (\widehat{\mathbf{y}} - \mathbf{y})^2$$

LASSO Regression

Ridge Regression

$$\widehat{\beta}_{ridge} = (\mathbf{X}^{\mathbf{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathbf{T}}\mathbf{y}$$

and

$$\widehat{\beta}_{ridge} = \arg\min_{\widehat{\beta}} \sum(\widehat{\mathbf{y}} - \mathbf{y})^2 + \lambda \sum \beta^2$$

LASSO Regression

$$\widehat{\beta}_{LASSO} = \arg\min_{\widehat{\beta}} \frac{1}{2}\sum(\widehat{\mathbf{y}} - \mathbf{y})^2 + \lambda \sum |\beta|,$$

where

$$\lambda \sum_{j=1}^{p} |\beta_j| = \lambda|\beta_j| + \lambda \sum_{k\neq j}^{p} |\beta_k|$$

LASSO Regression

Coordinate Descent

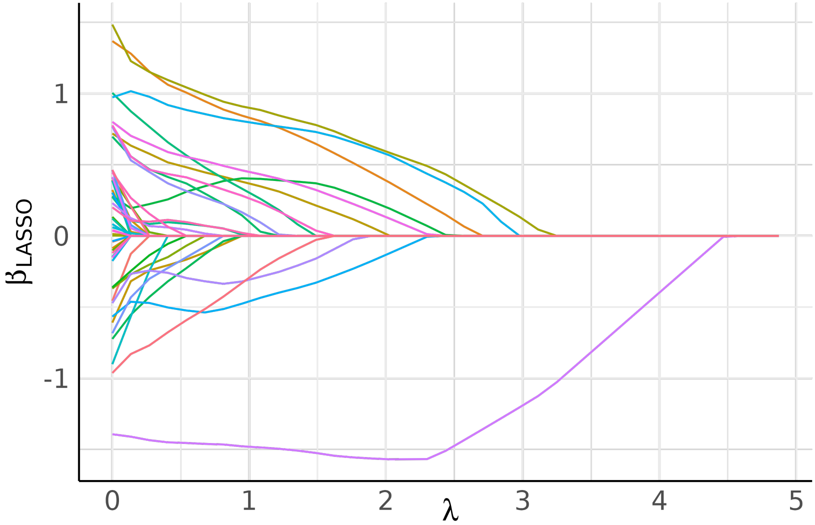
$$S(\rho_j, \lambda) = \beta_j = \begin{cases} \frac{\rho_j + \lambda}{z_j} & \text{for } \rho_j < -\lambda \\ 0 & \text{for } -\lambda \leq \rho_j \leq \lambda \\ \frac{\rho_j - \lambda}{z_j} & \text{for } \rho_j > \lambda \end{cases}$$

where $z_j = 1$ when the data are normalized (so you can ignore it)

Technical references

- coordinate descent
- LASSO

LASSO Regression



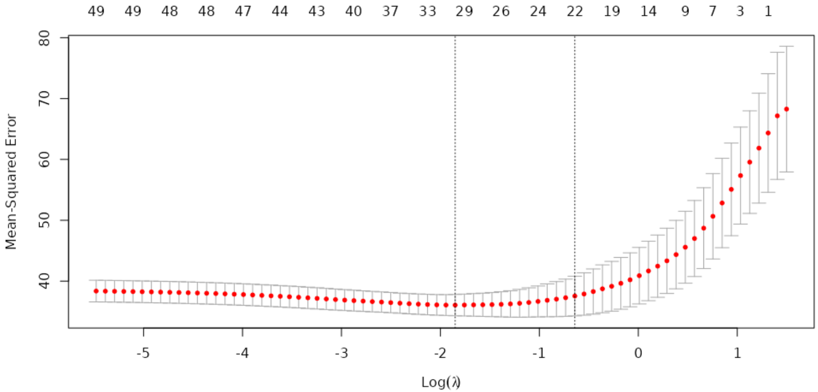
LASSO Regressionn

R Example

- The only change for LASSO regression is to use `alpha = 1`
- λ can similarly be chosen using cross-validation
- One major difference is that the LASSO often shrinks some β coefficients to zero (performing some feature selection on your behalf!)

Perform Cross-validation to Obtain λ

- Use the `ncds_sample.RData` dataset (don't forget to only keep `complete.cases`)
- Set a seed
- Using the following for your predictors and outcome:
  - Predictors: `ncds_sample[,2:51]`
  - Outcome: `ncds_sample[,"wem_well_being"]`
- Perform cross-validation LASSO regularization using 5-folds
- `print` and `plot` the output: What is the `min` λ?




```
    Lambda Index Measure    SE Nonzero
min 0.1571    37   36.06 1.751      33
1se 0.5266    24   37.55 3.260      22
```

Difference in Coefficients

```r
# Obtain coefficients of best LASSO
lasso_best <- glmnet(
  x = X, y = Y, family = "gaussian",
  alpha = 1, lambda = lasso_cv$lambda.min
)

# Standard linear model
standard_lm <- lm(wem_well_being ~ ., data = ncds_sample)

# Compute difference between standard and ridge coefficients
mean(abs(coef(lasso_best)[-1] - coef(standard_lm)[-1]))
```

```
[1] 0.1741861
```

```r
range(abs(coef(lasso_best)[-1] - coef(standard_lm)[-1]))
```

```
[1] 0.003204905 0.445243491
```

Generalizability?

Predict New Sample

- Load in the `ncds_test.RData` dataset (don't forget to only keep `complete.cases`)
- Get predictions from standard and LASSO models
    - Standard: `predict(standard_lm, newdata = ncds_test)`
    - LASSO: `predict(lasso_best, newx = X_test)`
- Compute RMSE for both standard and LASSO model
- Which model generalized better?

New Sample:

```
Standard RMSE: 0.3379202
```

```
LASSO RMSE: 0.303368
```

Did we generalize better?

New Sample:

```
Standard RMSE: 0.3379202
```

```
LASSO RMSE: 0.303368
```

Did we generalize better?

What about the original sample?

```
Standard RMSE: 0.00000000000003376473
```

```
LASSO RMSE: 0.000000000000002264283
```

How about R^2?

	Train	Test	# of Predictors
Standard	0.609	0.447	50
Ridge	0.578	0.490	50
LASSO	0.591	0.494	33

Do you prefer standard, ridge, or LASSO?

At Home Activity

- Select a binary (dichotomous) outcome of interest (see `ncds_codebook.xlsx` for descriptions of variables)
- Use the personality variables (columns 2-51) to predict your outcome
- Perform standard, ridge, and LASSO **logistic** regression on the `ndcs_sample.RData` and predict `ndcs_test.RData`
- Discuss which method you would prefer and why

Readings for Next Week

Readings

- ESL Chapters: 11.1-11.5
- HML: Chapter 13
- 3Blue1Brown YouTube
- Brilliant Wiki on Backpropagation

Optional

- Urban and Gates - 2021
- Smith - 2018
- Optimization for Deep Learning