# regls: regularized least squares in gretl

Allin Cottrell

Wake Forest University

Gretl virtual conference, 2021

# Preliminary remarks

I will exploit my privilege to talk first about the state of the gretl project, since there are some nice things to report!

# regls: an instance of hybrid design

Like our dbnomics and geoplot addons, regls has a bybrid design.

Combination of hansl and C components.

C for speed; hansl for brevity, transparency and ease of maintenance.

Briefly visit some examples...

# regls: an instance of hybrid design

Like our dbnomics and geoplot addons, regls has a bybrid design.

Combination of hansl and C components.

C for speed; hansl for brevity, transparency and ease of maintenance.

Briefly visit some examples...

## regls: an instance of hybrid design

Like our dbnomics and geoplot addons, regls has a bybrid design.

Combination of hansl and C components.

C for speed; hansl for brevity, transparency and ease of maintenance.

Briefly visit some examples...

## regls: an instance of hybrid design

Like our dbnomics and geoplot addons, regls has a bybrid design.

Combination of hansl and C components.

C for speed; hansl for brevity, transparency and ease of maintenance.

Briefly visit some examples...

hansl
C

# Regularized least squares

- Why? Danger of over-fitting, focus on out-of-sample prediction
- What methods? LASSO, Ridge regression, Elastic net
- What limitations? No generalized linear models at present

## Regularized least squares

- Why? Danger of over-fitting, focus on out-of-sample prediction
- What methods? LASSO, Ridge regression, Elastic net
- What limitations? No generalized linear models at present

## Regularized least squares

- Why? Danger of over-fitting, focus on out-of-sample prediction
- What methods? LASSO, Ridge regression, Elastic net
- What limitations? No generalized linear models at present

# LASSO 1

I will concentrate on LASSO, because of

- my time limitation
- its computational interest
- its effectiveness
- the relative transparency of the regularization factor

# LASSO 1

I will concentrate on LASSO, because of

- my time limitation
- its computational interest
- its effectiveness
- the relative transparency of the regularization factor

# LASSO 1

I will concentrate on LASSO, because of

- my time limitation
- its computational interest
- its effectiveness
- the relative transparency of the regularization factor

# LASSO 1

I will concentrate on LASSO, because of

- my time limitation
- its computational interest
- its effectiveness
- the relative transparency of the regularization factor

# LASSO 2

We use the parameterization of Boyd *et al* (2010), with objective:

$$\min_{\hat{\beta}} \quad \frac{1}{2} \sum_{i=1}^{n} (y_i - X_i \hat{\beta})^2 + \lambda \sum_{j=1}^{k} |\hat{\beta}_j|$$

$n$ = number of observations
$k$ = number of candidate regressors (columns of $X$)
$\lambda \geq 0$ is the regularization hyperparameter

$\lambda = 0$ gives plain OLS. And

$$\lambda_{\max} = \|X' y\|_\infty$$

drives all elements of $\hat{\beta}$ to zero.

Key regls control variable: $s = \lambda / \lambda_{\max}$

# Scripting basics

regls function signature:

```
bundle b = regls(series y, list X,
                 const bundle params[null])
```

The params bundle can contain a lot of controls, but all have default values.

Minimal directive for invoking cross validation:

```
bundle b = regls(y, X, _(xvalidate=1))
```

See the doc for details! And then there's the GUI...

# Scripting basics

regls function signature:

```
bundle b = regls(series y, list X,
                 const bundle params[null])
```

The `params` bundle can contain a lot of controls, but all have default values.

Minimal directive for invoking cross validation:

```
bundle b = regls(y, X, _(xvalidate=1))
```

See the doc for details! And then there's the GUI...

# Scripting basics

regls function signature:

```
bundle b = regls(series y, list X,
                 const bundle params[null])
```

The `params` bundle can contain a lot of controls, but all have default values.

Minimal directive for invoking cross validation:

```
bundle b = regls(y, X, _(xvalidate=1))
```

See the doc for details! And then there's the GUI...

gretl

# LASSO 3

Two comparisons of interest:

1. Numerical algorithm to pick the $\hat{\beta}$ that minimizes the LASSO criterion. We compare ADMM (Boyd *et al.*) with CCD (glmnet).
2. Alternative cross validation methodologies.

ADMM = Alternating Direction Method of Multipliers
CCD = Cyclical Coordinate Descent

Both algorithms are available in regls.

Full details on these points can be found in the Appendices to the regls documentation.

# LASSO 3

Two comparisons of interest:

1. Numerical algorithm to pick the $\hat{\beta}$ that minimizes the LASSO criterion. We compare ADMM (Boyd *et al.*) with CCD (glmnet).
2. Alternative cross validation methodologies.

ADMM = Alternating Direction Method of Multipliers
CCD = Cyclical Coordinate Descent

Both algorithms are available in regls.

Full details on these points can be found in the Appendices to the regls documentation.

# LASSO 3

Two comparisons of interest:

1. Numerical algorithm to pick the $\hat{\beta}$ that minimizes the LASSO criterion. We compare ADMM (Boyd *et al.*) with CCD (glmnet).
2. Alternative cross validation methodologies.

ADMM = Alternating Direction Method of Multipliers
CCD = Cyclical Coordinate Descent

Both algorithms are available in regls.

Full details on these points can be found in the Appendices to the regls documentation.

# ADMM vs CCD accuracy experiment: setup

Use the US murder rates dataset supplied with regls: `murdPerPop` as dependent variable and 101 candidate regressors. Data pre-standardized in this experiment.

Perform LASSO estimation using 20 values of $\lambda$, 800 observations. Record the minimized LASSO criteria, $c_i$, $i = 1, 2, \ldots, 20$.

Take ADMM as baseline; compare with CCD starting at its default tolerance and progressively tightening.

Comparative measures:

- Euclidean distance between results: $\sqrt{(dc' dc)}$, where $dc$ is the difference vector $c_{\text{admm}} - c_{\text{ccd}}$.
- Relative execution time: CCD/ADMM.

# ADMM vs CCD accuracy experiment: setup

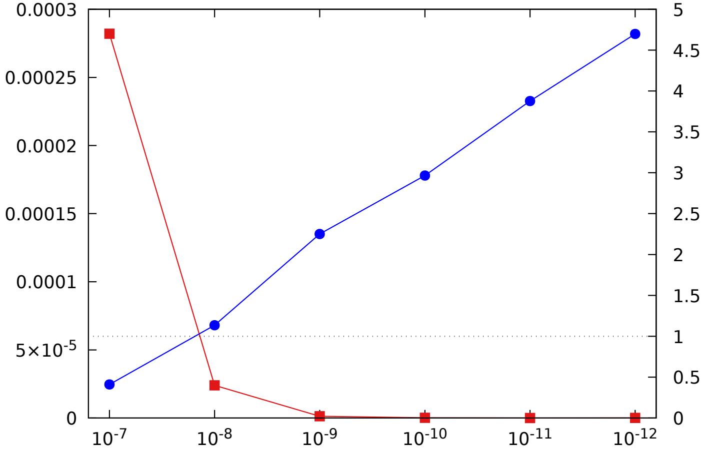
Use the US murder rates dataset supplied with regls: `murdPerPop` as dependent variable and 101 candidate regressors. Data pre-standardized in this experiment.

Perform LASSO estimation using 20 values of $\lambda$, 800 observations. Record the minimized LASSO criteria, $c_i$, $i = 1, 2, \ldots, 20$.

Take ADMM as baseline; compare with CCD starting at its default tolerance and progressively tightening.

Comparative measures:

- Euclidean distance between results: $\sqrt{(dc'\,dc)}$, where $dc$ is the difference vector $c_{\mathrm{admm}} - c_{\mathrm{ccd}}$.
- Relative execution time: CCD/ADMM.




LASSO estimation: CCD performance relative to ADMM. CCD tolerance on *x*-axis; Euclidean distance between results in red (left); relative execution time in blue (right).

# Cross validation methodology

This differs between regls and the glmnet package for R.

- regls: standardization and computation of $\lambda$-sequence are done once, using the entire training sample.
- glmnet: standardization and computation of $\lambda$-sequence are done per-fold, using the sample complementary to the given fold.

It is not clear *a priori* which method will produce better results.

But the results should not differ by much if the training data are relatively homogeneous.

# Cross validation methodology

This differs between regls and the glmnet package for R.

- regls: standardization and computation of $\lambda$-sequence are done once, using the entire training sample.
- glmnet: standardization and computation of $\lambda$-sequence are done per-fold, using the sample complementary to the given fold.

It is not clear *a priori* which method will produce better results.

But the results should not differ by much if the training data are relatively homogeneous.

# Cross validation methodology

This differs between regls and the glmnet package for R.

- regls: standardization and computation of $\lambda$-sequence are done once, using the entire training sample.
- glmnet: standardization and computation of $\lambda$-sequence are done per-fold, using the sample complementary to the given fold.

It is not clear *a priori* which method will produce better results.

But the results should not differ by much if the training data are relatively homogeneous.

# Cross validation methodology

This differs between regls and the glmnet package for R.

- regls: standardization and computation of $\lambda$-sequence are done once, using the entire training sample.
- glmnet: standardization and computation of $\lambda$-sequence are done per-fold, using the sample complementary to the given fold.

It is not clear *a priori* which method will produce better results.

But the results should not differ by much if the training data are relatively homogeneous.

# Cross validation experiment

- Dataset 1: murder rates and covariates for US localities, $n = 2215$, $k = 102$.
- Dataset 2: white wine quality and physico-chemical covariates, $n = 4898$, $k = 12$ (78 after adding squares and interactions).

At each of 2000 iterations:

- Randomize the order of the entire dataset.
- Use the first $N$ observations for training and the next $M$ for testing. (Dataset 1: $N = 1200$, $M = 200$; Dataset 2: $N = 1500$, $M = 500$.)
- Perform cross validation with 10 folds.
- Select optimal $\lambda$ on the "one standard error" rule.
- Predict for the testing observations and calculate $R^2 = 1 - \sum(y - \hat{y})^2 / \sum(y - \bar{y})^2$.

# Cross validation experiment

- Dataset 1: murder rates and covariates for US localities, $n = 2215$, $k = 102$.
- Dataset 2: white wine quality and physico-chemical covariates, $n = 4898$, $k = 12$ (78 after adding squares and interactions).

At each of 2000 iterations:

- Randomize the order of the entire dataset.
- Use the first $N$ observations for training and the next $M$ for testing. (Dataset 1: $N = 1200$, $M = 200$; Dataset 2: $N = 1500$, $M = 500$.)
- Perform cross validation with 10 folds.
- Select optimal $\lambda$ on the "one standard error" rule.
- Predict for the testing observations and calculate $R^2 = 1 - \sum(y - \hat{y})^2 / \sum(y - \bar{y})^2$.

# Cross validation experiment

- Dataset 1: murder rates and covariates for US localities, $n = 2215$, $k = 102$.
- Dataset 2: white wine quality and physico-chemical covariates, $n = 4898$, $k = 12$ (78 after adding squares and interactions).

At each of 2000 iterations:

- Randomize the order of the entire dataset.
- Use the first $N$ observations for training and the next $M$ for testing. (Dataset 1: $N = 1200$, $M = 200$; Dataset 2: $N = 1500$, $M = 500$.)
- Perform cross validation with 10 folds.
- Select optimal $\lambda$ on the "one standard error" rule.
- Predict for the testing observations and calculate $R^2 = 1 - \sum(y - \hat{y})^2 / \sum(y - \bar{y})^2$.

# Cross validation experiment

- Dataset 1: murder rates and covariates for US localities, $n = 2215$, $k = 102$.
- Dataset 2: white wine quality and physico-chemical covariates, $n = 4898$, $k = 12$ (78 after adding squares and interactions).

At each of 2000 iterations:

- Randomize the order of the entire dataset.
- Use the first $N$ observations for training and the next $M$ for testing. (Dataset 1: $N = 1200$, $M = 200$; Dataset 2: $N = 1500$, $M = 500$.)
- Perform cross validation with 10 folds.
- Select optimal $\lambda$ on the "one standard error" rule.
- Predict for the testing observations and calculate $R^2 = 1 - \sum(y - \hat{y})^2 / \sum(y - \bar{y})^2$.

# Cross validation experiment

- Dataset 1: murder rates and covariates for US localities, $n = 2215$, $k = 102$.
- Dataset 2: white wine quality and physico-chemical covariates, $n = 4898$, $k = 12$ (78 after adding squares and interactions).

At each of 2000 iterations:

- Randomize the order of the entire dataset.
- Use the first $N$ observations for training and the next $M$ for testing. (Dataset 1: $N = 1200$, $M = 200$; Dataset 2: $N = 1500$, $M = 500$.)
- Perform cross validation with 10 folds.
- Select optimal $\lambda$ on the "one standard error" rule.
- Predict for the testing observations and calculate $R^2 = 1 - \sum(y - \hat{y})^2 / \sum(y - \bar{y})^2$.

# Cross validation experiment

- Dataset 1: murder rates and covariates for US localities, $n = 2215$, $k = 102$.
- Dataset 2: white wine quality and physico-chemical covariates, $n = 4898$, $k = 12$ (78 after adding squares and interactions).

At each of 2000 iterations:

- Randomize the order of the entire dataset.
- Use the first $N$ observations for training and the next $M$ for testing. (Dataset 1: $N = 1200$, $M = 200$; Dataset 2: $N = 1500$, $M = 500$.)
- Perform cross validation with 10 folds.
- Select optimal $\lambda$ on the "one standard error" rule.
- Predict for the testing observations and calculate $R^2 = 1 - \sum(y - \hat{y})^2 / \sum(y - \bar{y})^2$.

Out-of-sample $R^2$, comparative statistics, 2000 trials

| | mean | s.d. | s.e.(mean) | median |
|---|---|---|---|---|
| glmnet | 0.4724 | 0.1518 | 0.0034 | 0.4881 |
| regls CCD | 0.4954 | 0.1545 | 0.0035 | 0.5118 |
| regls ADMM | 0.4984 | 0.1608 | 0.0036 | 0.5172 |

Paired-difference tests and correlations

| | $\lvert z \rvert$ | $\rho$ |
|---|---|---|
| glmnet, regls CCD | 20.6 | 0.946 |
| glmnet, regls ADMM | 21.6 | 0.942 |
| regls CCD, regls ADMM | 8.4 | 0.996 |

Maybe easier to visualize. . .

Out-of-sample $R^2$, comparative statistics, 2000 trials

| | mean | s.d. | s.e.(mean) | median |
|---|---|---|---|---|
| glmnet | 0.4724 | 0.1518 | 0.0034 | 0.4881 |
| regls CCD | 0.4954 | 0.1545 | 0.0035 | 0.5118 |
| regls ADMM | 0.4984 | 0.1608 | 0.0036 | 0.5172 |

Paired-difference tests and correlations

| | $\|z\|$ | $\rho$ |
|---|---|---|
| glmnet, regls CCD | 20.6 | 0.946 |
| glmnet, regls ADMM | 21.6 | 0.942 |
| regls CCD, regls ADMM | 8.4 | 0.996 |

Maybe easier to visualize. . .

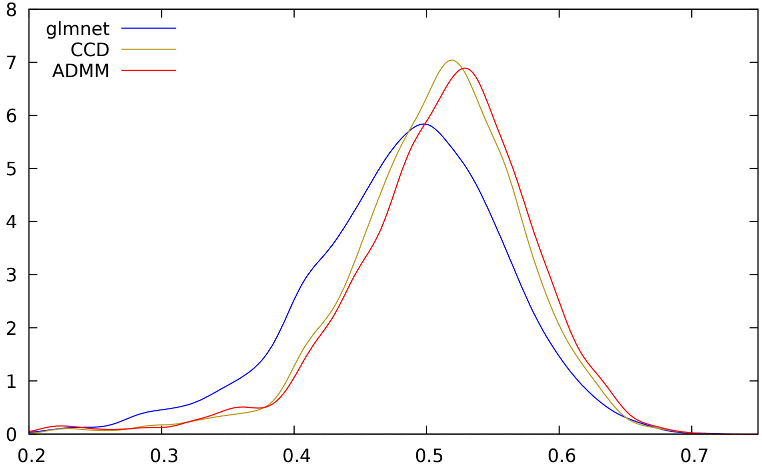
Out-of-sample $R^2$, comparative statistics, 2000 trials

| | mean | s.d. | s.e.(mean) | median |
|---|---|---|---|---|
| glmnet | 0.4724 | 0.1518 | 0.0034 | 0.4881 |
| regls CCD | 0.4954 | 0.1545 | 0.0035 | 0.5118 |
| regls ADMM | 0.4984 | 0.1608 | 0.0036 | 0.5172 |

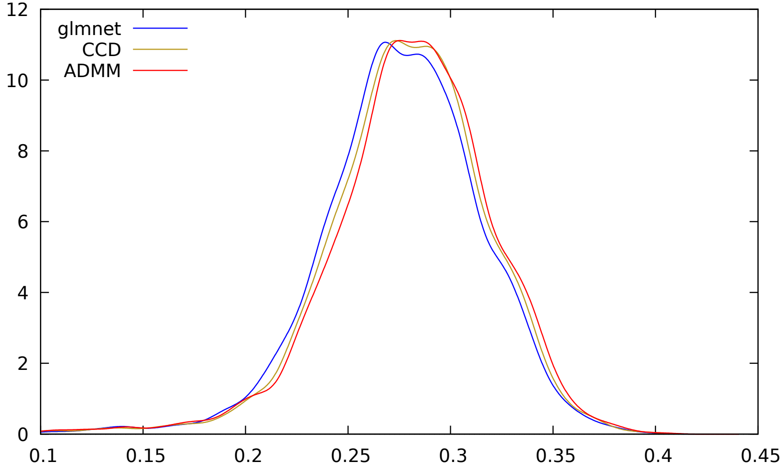
Paired-difference tests and correlations

| | $\lvert z \rvert$ | $\rho$ |
|---|---|---|
| glmnet, regls CCD | 20.6 | 0.946 |
| glmnet, regls ADMM | 21.6 | 0.942 |
| regls CCD, regls ADMM | 8.4 | 0.996 |

Maybe easier to visualize. . .




Estimated densities for out of sample $R^2$, murder rates data




Estimated densities for out of sample $R^2$, wine quality data

# Dataset heterogeneity?

For $K$ trials indexed by $i$ and $F$ folds indexed by $j$, $\bar{y}$ = sample mean and $s$ = sample standard deviation of the dependent variable:

$$H_\mu = K^{-1} \sum_{i=1}^{K} \sum_{j=1}^{F} |\bar{y}_{ij} - \bar{y}_i| / |\bar{y}_i|$$

$$H_\sigma = K^{-1} \sum_{i=1}^{K} \sum_{j=1}^{F} |s_{ij} - s_i| / s_i$$

| | $H_\mu$ | $H_\sigma$ |
|---|---|---|
| Murder rates data | 0.12059 | 0.15032 |
| Wine quality data | 0.01039 | 0.05089 |

# Dataset heterogeneity?

For $K$ trials indexed by $i$ and $F$ folds indexed by $j$, $\bar{y}$ = sample mean and $s$ = sample standard deviation of the dependent variable:

$$H_\mu = K^{-1} \sum_{i=1}^{K} \sum_{j=1}^{F} |\bar{y}_{ij} - \bar{y}_i| / |\bar{y}_i|$$

$$H_\sigma = K^{-1} \sum_{i=1}^{K} \sum_{j=1}^{F} |s_{ij} - s_i| / s_i$$

| | $H_\mu$ | $H_\sigma$ |
|---|---|---|
| Murder rates data | 0.12059 | 0.15032 |
| Wine quality data | 0.01039 | 0.05089 |