Data Mining

Linear Model Selection

Santiago Alonso-Díaz

Tecnólogico de Monterrey
EGADE, Business School

Photo: Dalle2

Good ideas

What metrics do we have for good ideas in crowd-sourcing sites that compete for resources? Are all important? Bell et al., 2024

- Prototypical ideas
  - Var. 1 (word typicality in Google search)
  - Var. 2 (node typicality from a semantic network in Google search)
  - Var. 3 (other e.g. KS distances between documents)
- Atypical ideas among proposals
  - Var. 4 (word typicality in the available proposals)
  - Var. 5 (topic typicality in the available proposals)
- Ideators with good connections
  - Var. 6 (degree of connections)
  - Var. 7 (transitivity of connections)
  - Var. 8 (constraints of connections)

With automated model selection, in this case LASSO, we reduced the number of explanatory variables (DV: short listed (yes/no))

Table: Selected variables via LASSO (Bell et al., 2024)

	Standardized Coefficient
Intercept	-3.50
Ideators Var. 1	-0.19
Prototypical Var. 3	-0.08
Atypicality Var. 2	-0.07
Control (% shortlisted, contest-level)	0.10

Managers/investors can decide model complexity

The authors do more analyses and models (e.g. two step approach) and provide this counsel to pick good crowd-sourcing ideas:

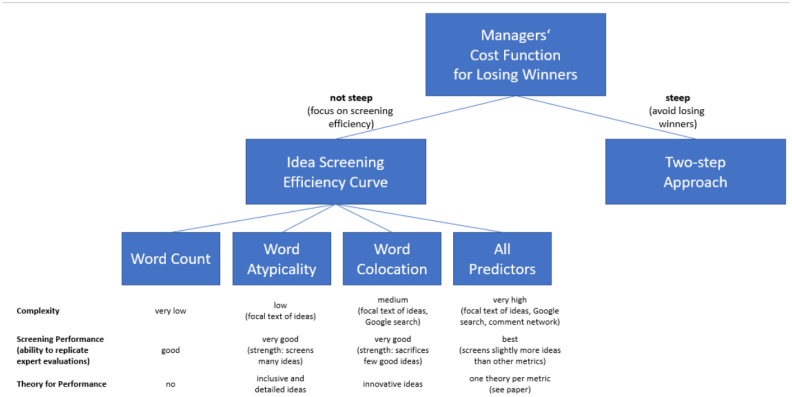


Figure: Strategy recommendations by Bell et al., 2024

Some criteria

- Fit to data (low bias)
- Good predictions (low variance)
- Easy to explain (low complexity)

Table of contents

Subset and stepwise selection (James et al., 2023)

Subset selection

Brute force: Test all possible combinations of predictors and pick the best according some criteria. For instance R^2,

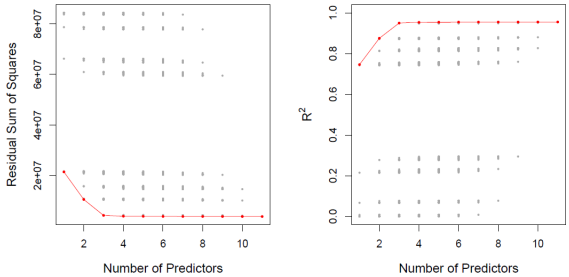


FIGURE 6.1. *For each possible model containing a subset of the ten predictors in the* `Credit` *data set, the RSS and* R^2 *are displayed. The red frontier tracks the best model for a given number of predictors, according to RSS and* R^2*. Though*

Figure: More than 2 or 3 predictors seems unnecessary (James et al., 2023)

Stepwise selection

Less brute force (but still brute). There are different versions.

- Forward selection
  - Start with a model with no predictors
  - At each step add one predictor, the one that improves a goodness metric (e.g. R^2, AIC, cross validation, other)
  - Repeat and stop until a rule of improvement
- Backward selection
  - Start with a model with ALL the predictors
  - At each step drop the worst predictor according to some goodness metric (e.g. R^2, AIC, cross validation, other)
  - Repeat and stop until a rule of improvement

Comparison of subset and forward selection

# Variables	Best subset	Forward stepwise
One	`rating`	`rating`
Two	`rating`, `income`	`rating`, `income`
Three	`rating`, `income`, `student`	`rating`, `income`, `student`
Four	`cards`, `income`	`rating`, `income`,
	`student`, `limit`	`student`, `limit`

TABLE 6.1. *The first four selected models for best subset selection and forward stepwise selection on the* `Credit` *data set. The first three models are identical but the fourth models differ.*

Figure: (James et al., 2023)

Some goodness metrics

- R^2 (but it doesn't penalize complexity)
- Adjusted R^2 (penalizes complexity)
- Information Criterions (AIC, BIC, WAIC; max. likelihood and penalizes complexity)
- C_p (RSS penalized by complexity)
- Validation, Cross-Validation

Comparison of goodness metrics

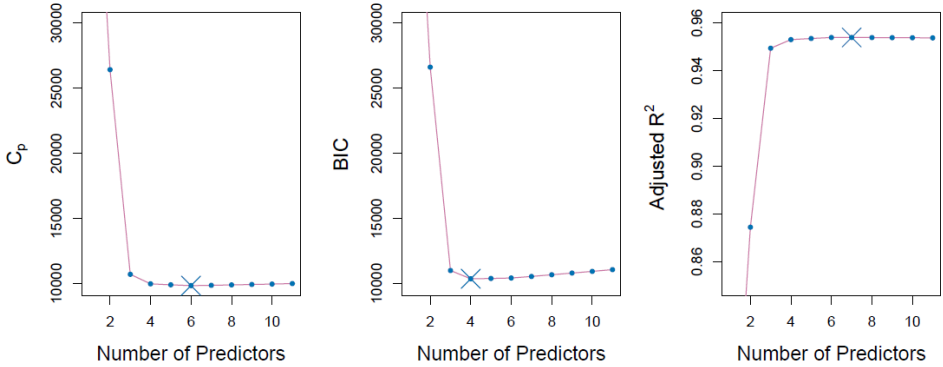


Figure: Credit Data. BIC, C_p, lower better. R^2, higher better. (James et al., 2023)

Comparison of goodness metrics

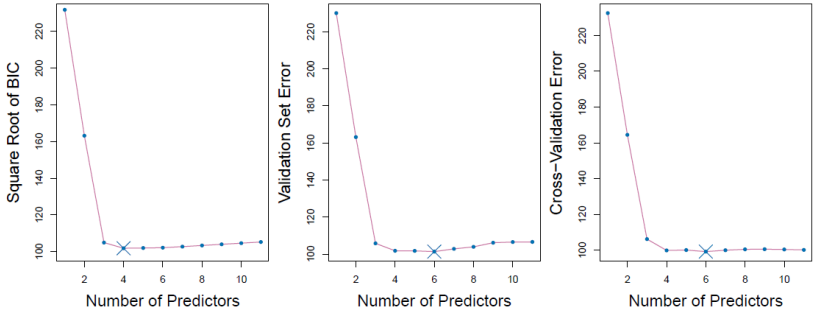


FIGURE 6.3. *For the* `Credit` *data set, three quantities are displayed for the best model containing* d *predictors, for* d *ranging from* 1 *to* 11. *The overall* best *model, based on each of these quantities, is shown as a blue cross.* Left: *Square root of BIC.* Center: *Validation set errors.* Right: *Cross-validation errors.*

Figure: Credit Data. (James et al., 2023)

Drawbacks of subset or stepwise procedures

- Investigator degrees of freedom (e.g. goodness measure, thresholds for stopping)
- Theory blind. It may drop relevant theoretical variables.
- Too dependent on data characteristics. New data, different parameters.

Shrinkage/Regularization (James et al., 2023)

General intuition

Penalize number of parameters in the cost function.
The original RSS (to minimize):

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

The intuition of regularization is this:

$$RSS_{regularized} = RSS + \text{penalty per each } \beta$$

Note that each additional β hinders the minimization i.e. it would be good that some β go to zero.

Ridge regression

$$\text{RSS}_{\text{ridge}} = RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

λ is a free tuning parameter obtainable, for instance, with cross validation. It determines how much we want coefficients to shrink. If too high, they go to zero.

As λ grows, βs shrink

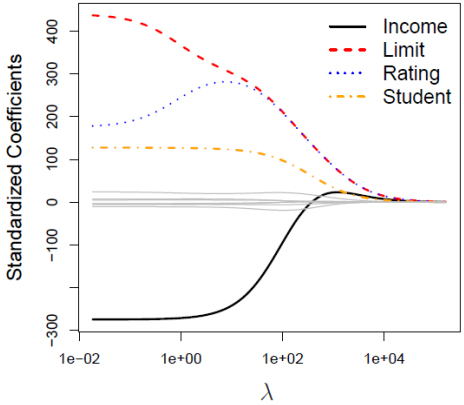


Figure: Credit Data. Ridge regression shrinks predictors, some approach zero faster (but never zero) (James et al., 2023)

As λ grows, bias goes up, variance down

Ridge regression modulates the bias (describe current data)-variance (predict new data) trade-off

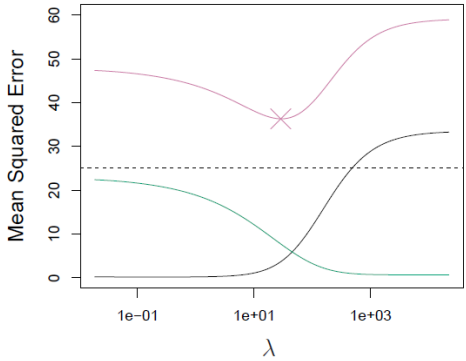


Figure: Credit Data. Black $bias^2$. Green variance. Purple MSE on test data. (James et al., 2023)

Lasso regression

Ridge never pulls βs to zero. Lasso does i.e. it effectively drops bad predictors:

$$\text{RSS}_{\text{Lasso}} = RSS + \lambda \sum_{j=1}^{p} |\beta_j|$$

Note that we use ℓ_1 i.e. absolute value. Again, λ is a free parameter obtainable via cross-validation.

βs to zero

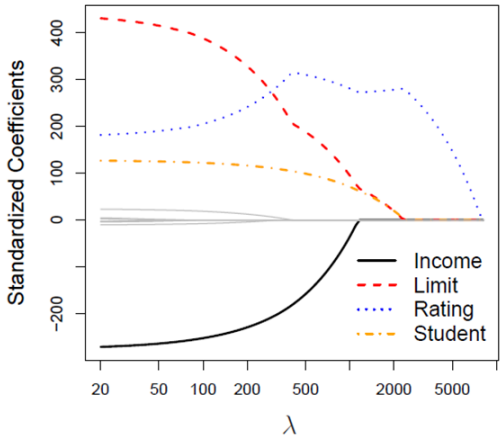


Figure: Credit Data. A large λ, coeff. disappear i.e. feature selection (James et al., 2023)

Why lasso, and not ridge, sets zeros?

Lasso and ridge are equivalent to this. βs have to be smaller than a value s.

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq s \tag{6.8}$$

and

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 \leq s, \tag{6.9}$$

Why lasso, and not ridge, sets zeros?

Geometrically, for lasso, the RSS (red) can in principle touch zeros and still respect the restriction (blue). For ridge, is harder due to the curvature.

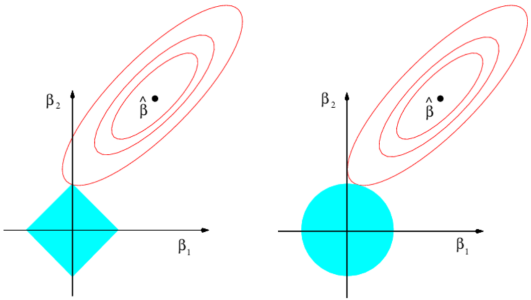


FIGURE 6.7. *Contours of the error and constraint functions for the lasso* (left) *and ridge regression* (right). *The solid blue areas are the constraint regions,* $|\beta_1| + |\beta_2| \leq s$ *and* $\beta_1^2 + \beta_2^2 \leq s$, *while the red ellipses are the contours of the RSS.*

Figure: Example with two βs (James et al., 2023)

Tuning parameter λ

Obtainable via cross validation. For example:

- Pick one λ from a grid of λs.
- Pick a random train and test set.
- Fit the model with Lasso or Ridge.
- Estimate and save a goodness measure for test prediction.
- Repeat for all the grid.
- Pick the best λ i.e. with the best prediction measure.

Drawbacks of ridge and lasso

- Theory blind. It may drop relevant theoretical variables.

Dimension reduction (James et al., 2023)

Dimension Reduction

We have already seen the spirit of reduction of dimensions e.g. forward selection, lasso. In our initial example, we went from 8 to 3 variables:

Table: Selected variables via LASSO (Bell et al., 2024)

	Standardized Coefficient
Intercept	-3.50
Ideators Var. 1	-0.19
Prototypical Var. 3	-0.08
Atypicality Var. 2	-0.07
Control (% shortlisted, contest-level)	0.10

Dimension reduction via PCA

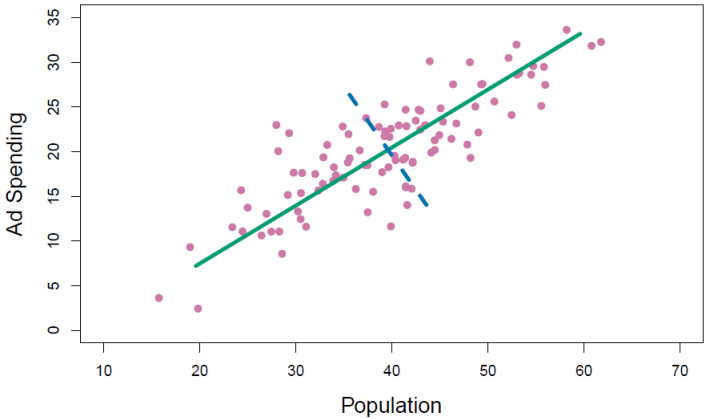


Figure: Example with two dims. We could decide to reduce/summarize the data with the first component (green): it is loaded with both dimensions. (James et al., 2023)

Dimension reduction via PCA

Each component is orthogonal and captures the largest possible variance (spread of points). No details, just the intuition. Later in Python we implement PCA.

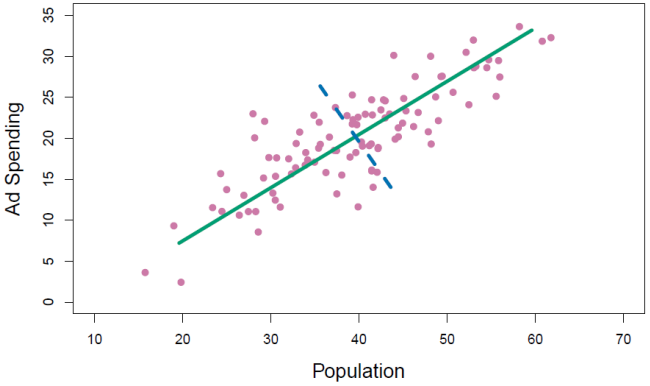


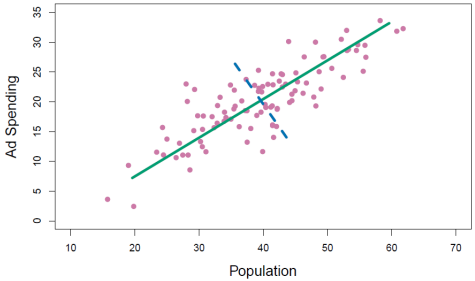
Figure: Example with two dims (James et al., 2023)

Dimension reduction via PCA

In formula, the first component Z_1 is:

$$Z_1 = 0.839 \times (pop_i - \bar{pop}) + 0.544(ad_i - \bar{ad})$$

We refer to 0.839 and 0.544 as loadings. We know the sample means, $\bar{pop}$ and $\bar{ad}$, thus for each pop_i and ad_i in the data we can calculate its position in the component i.e. we reduce two dimensions to one.



1st component

The first component is interesting because it has an additional interpretation (additional to capturing the most variance):

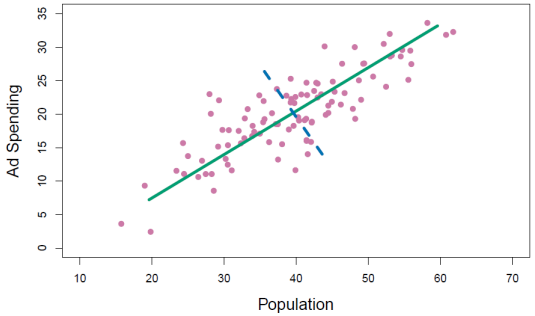
It is the closest to the data

For instance, if $Z_1 < 0$, that combination of population and ad expenses is below average

More components

We can do PCA with n variables in the data, and have up to n components. With two variables, we can estimate a second component (and no more):

$$Z_2 = 0.544 \times (pop_i - \bar{pop}) - 0.839(ad_i - \bar{ad})$$



PCA regression

We can use the desired components as variables in our regressions, conditional that we can interpret the components, via its loadings, in a sensical way. A broad overview:

- Standardized variables via z-scores
- Do PCA
- Select the number of components you consider relevant (or via cross validation)
- Calculate, for each component, the associated values for each observation in your data.
- Run your preferred regression with the reduced dimensions

PCA regression

Around 10 components is good (right), but almost all the variables (11), no dim. reduction.

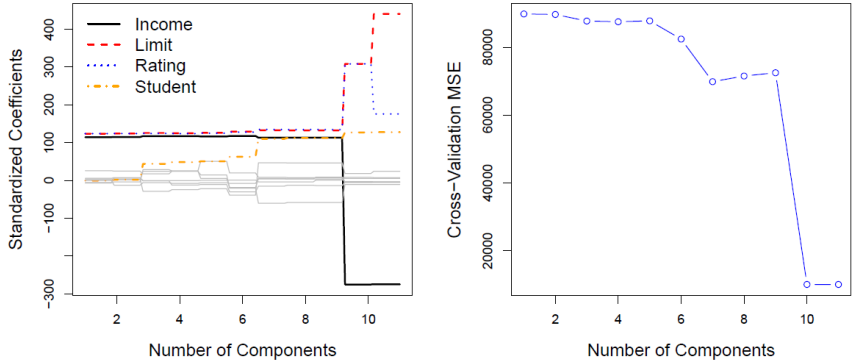


Figure: Credit dataset. Left: 11 variables but just 4 highlighted. Right: Cross validation (James et al., 2023)

Issues with PCA

- PCA is not feature selection. Each component is a linear sum of ALL features.
- Identification of different geometries (see figure)

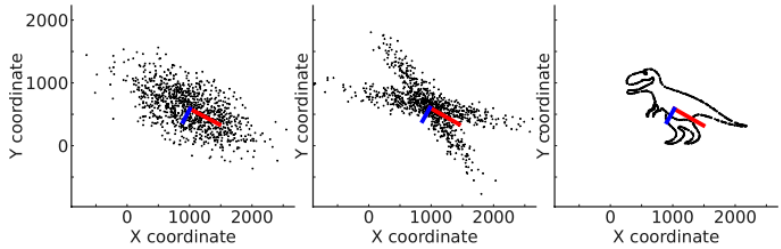


Figure: Similar PCA, different geometries (Dyer & Kording, 2023)

Issues with high dimensions (James et al., 2023)

High dimensions

Examples of n<predictors

- Predict a health variable of n = 1000 patients with age, sex, BMI, and 500.000 genes.
- Predict shopping patterns of n = 500 costumers with thousands of search terms in Google

Over-fitting with n$\leq$predictors

Bias - variance trade-off

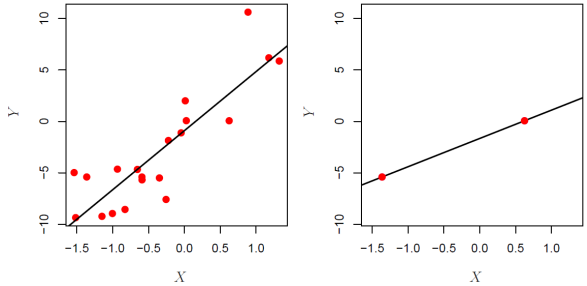


Figure: Right: 2 parameters (β_0, β_1) and 2 observations. The line explains 100% the data (zero bias) but will fail to predict new data because most likely is just for those two points (high variance). Left: more n reduces variance, but at a higher bias (which is fine for this case) (James et al., 2023)

Over-fitting with n$\leq$predictors

More parameters p lower bias but higher variance on test set

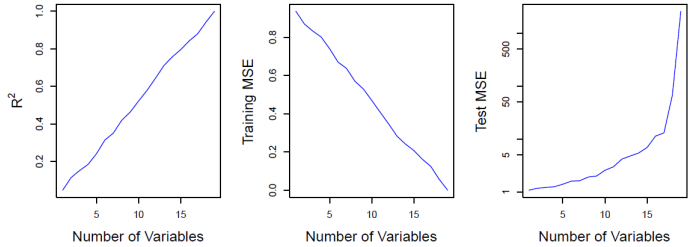


FIGURE 6.23. *On a simulated example with $n = 20$ training observations, features that are completely unrelated to the outcome are added to the model.* Left: *The R^2 increases to 1 as more features are included.* Center: *The training set MSE decreases to 0 as more features are included.* Right: *The test set MSE increases as more features are included.*

Figure: If n is small relative to p, R^2 is vague. Rather, use test MSE (James et al., 2023)

To Python

DM_Model_Selection.ipynb

References

Bell, J. J., Pescher, C., Tellis, G. J., & Füller, J. (2024). Can ai help in ideation? a theory-based model for idea screening in crowdsourcing contests. *Marketing Science*, *43*(1), 54–72.

Dyer, E. L., & Kording, K. (2023). Why the simplest explanation isn't always the best. *Proceedings of the National Academy of Sciences*, *120*(52), e2319169120.

James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). *An introduction to statistical learning: With applications in python*. Springer Nature.