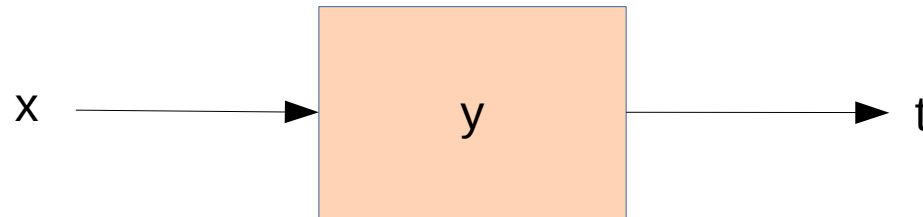


Introduction to Machine Learning

Lecture 5 Linear Models II

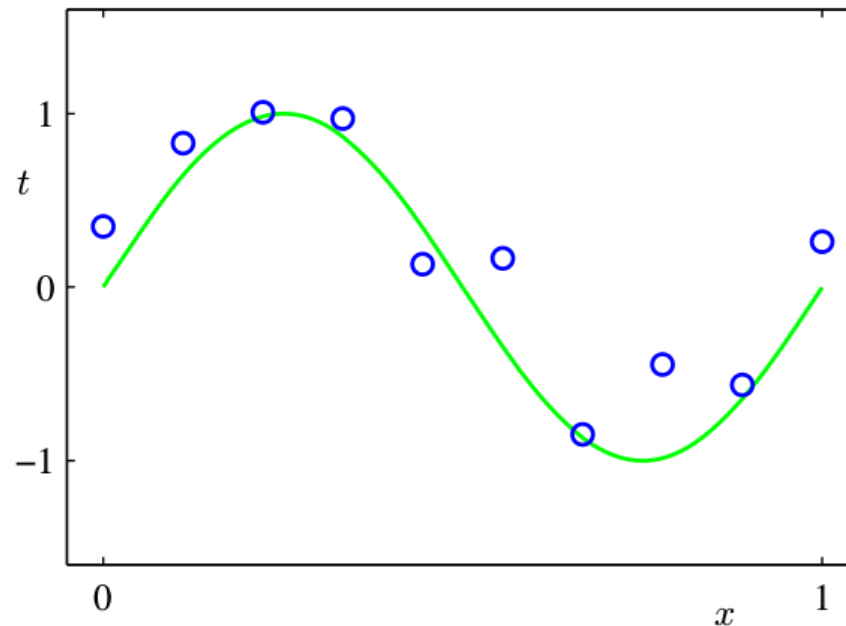
Goker Erdogan
26 – 30 November 2018
Pontificia Universidad Javeriana

Polynomial curve fitting example



- Given N samples (training set) of $\{x, t\}$, learn function y
 - So we can predict t for a new x

Plot of a training data set of $N = 10$ points, shown as blue circles, each comprising an observation of the input variable x along with the corresponding target variable t . The green curve shows the function $\sin(2\pi x)$ used to generate the data. Our goal is to predict the value of t for some new value of x , without knowledge of the green curve.



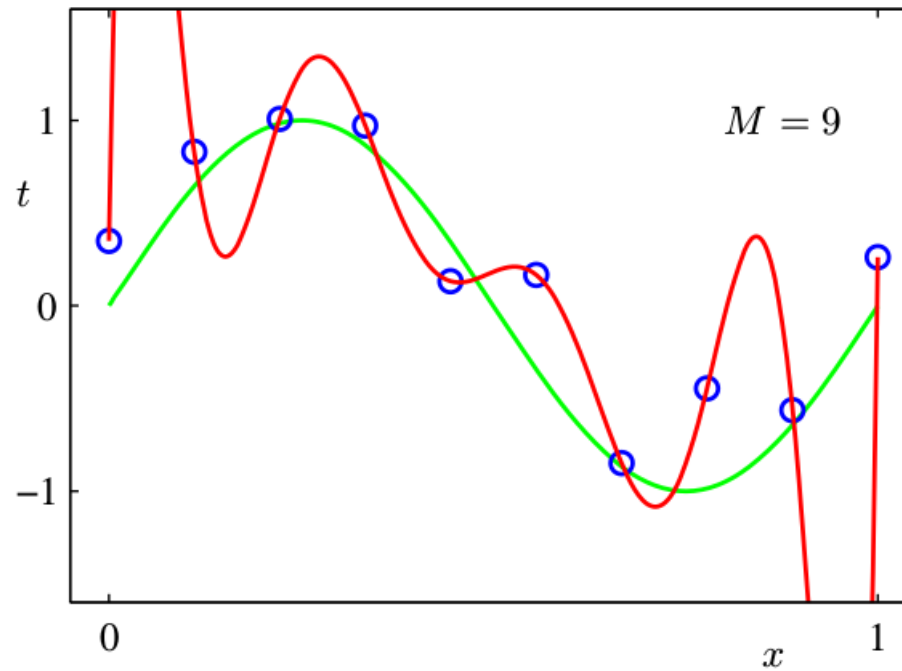


Table of the coefficients w^* for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M = 0$	$M = 1$	$M = 3$	$M = 9$
w_0^*	0.19	0.82	0.31	0.35
w_1^*		-1.27	7.99	232.37
w_2^*			-25.43	-5321.83
w_3^*			17.37	48568.31
w_4^*				-231639.30
w_5^*				640042.26
w_6^*				-1061800.52
w_7^*				1042400.18
w_8^*				-557682.99
w_9^*				125201.43

Regularization motivation

- As the **coefficients** get larger, the model becomes
 - More complex
 - Non-smooth
- Risk of **overfitting**
- Can we encourage coefficients to be **smaller**?
 - Give them a maximum **size/magnitude**.

$$\min_{\beta} (y - X\beta)^T (y - X\beta)$$

$$\text{s.t.} \quad \sum_{d=1}^D \beta_d^2 < C$$

$$(\text{or } \|\beta\|_2^2 < C)$$

Brief aside on norms

- Norm: a function that assigns a **strictly positive length or size** to each vector
- p -norm ($p \geq 1$)

$$||x||_p = \left(\sum_i |x_i| \right)^{1/p}$$

- $p=2$ (l_2 norm), Euclidean norm

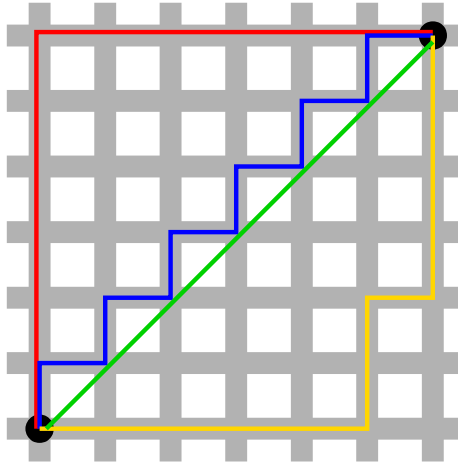
$$||x||_2 = \sqrt{\sum_i |x_i|^2}$$

- $p=1$ (l_1 norm), Taxicab (Manhattan) norm

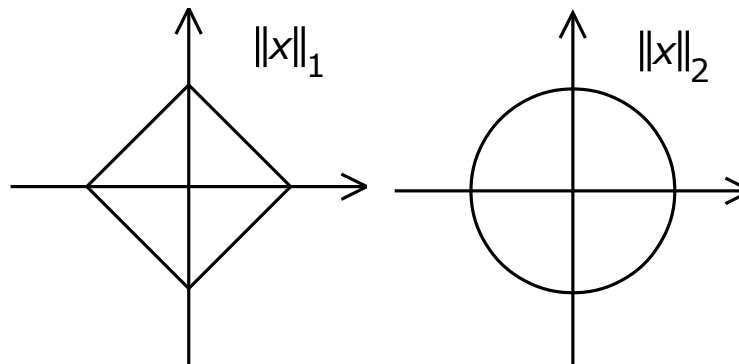
$$||x||_1 = \sum_i |x_i|$$

Brief aside on norms contd.

- Taxicab (Manhattan) norm

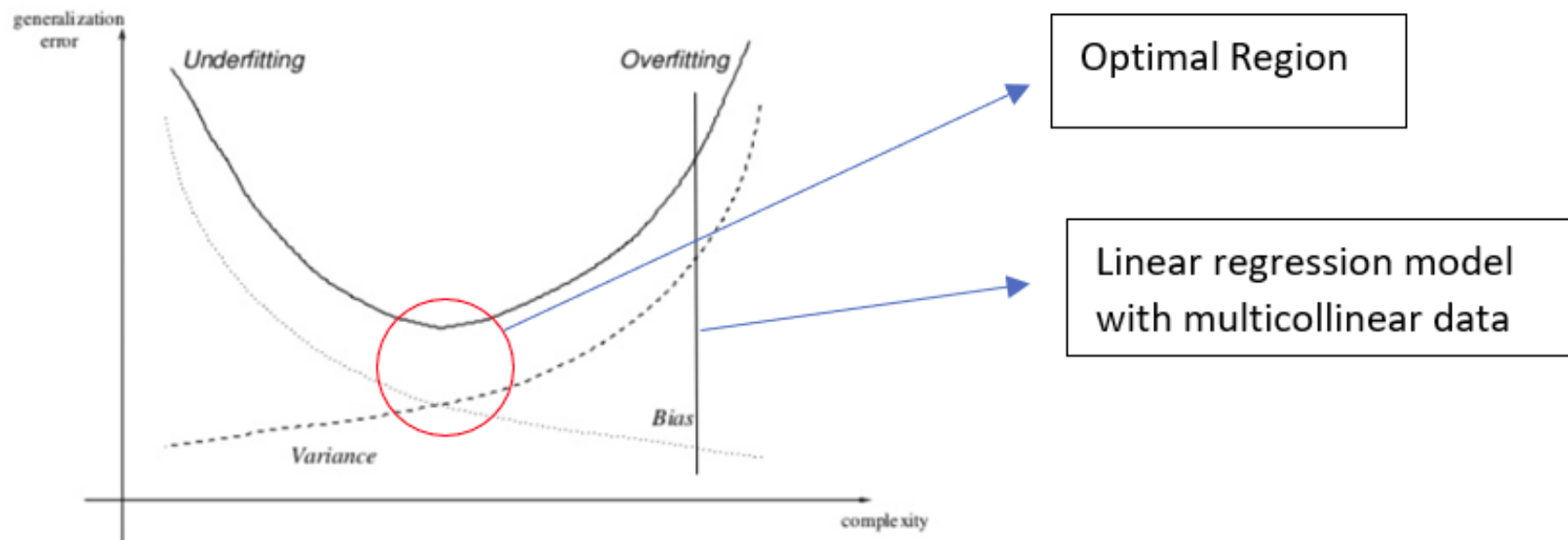


- Unit balls (circles) in L1 and L2



Regularization

- Introducing **additional information/assumptions** to
 - Solve an ill-posed problem
 - Prevent overfitting
- Reduces the variance substantially (with a slight increase in bias)



l_2 regularization

- Known as Ridge regression, weight decay, Tikhonov regularization
- Solve

$$\text{RSS} = \sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 .$$
$$\text{s.t.} \quad \sum_{p=1} \beta_p^2 < C$$

- Equivalently (add constraint as a penalty)

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

l_2 regularization

- Minimize wrt to β

$$\text{RSS}(\lambda) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta) + \lambda \beta^T \beta,$$

- There is a closed form solution

$$\hat{\beta}^{\text{ridge}} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y},$$

- Adding the identity matrix makes the system non-singular
- Compare with least-squares estimate

$$\hat{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}.$$

What does l_2 regularization do?

- Pushes β towards zero
- λ controls how much we shrink them
 - Pick according to test performance (e.g., cross-validation)

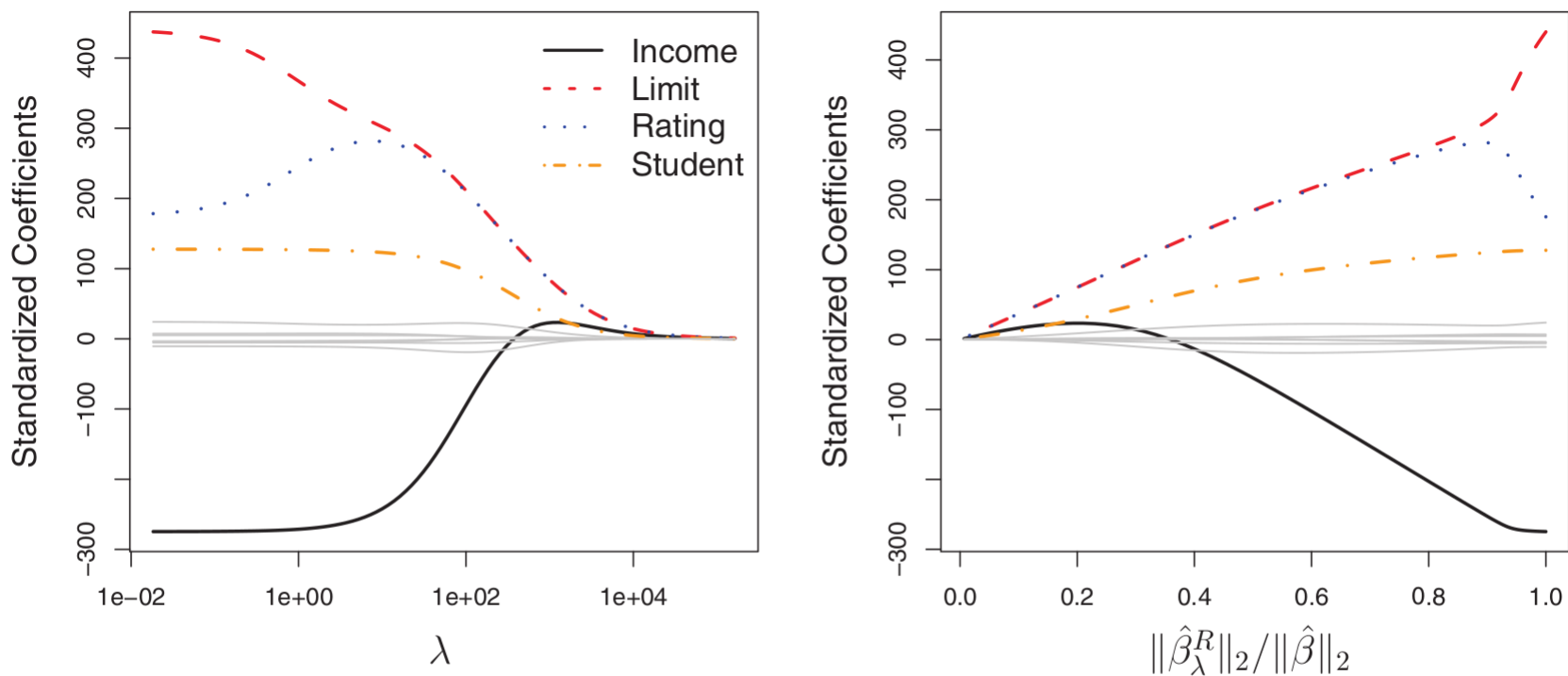


FIGURE 6.4. The standardized ridge regression coefficients are displayed for the **Credit** data set, as a function of λ and $\|\hat{\beta}_\lambda^R\|_2 / \|\hat{\beta}\|_2$.

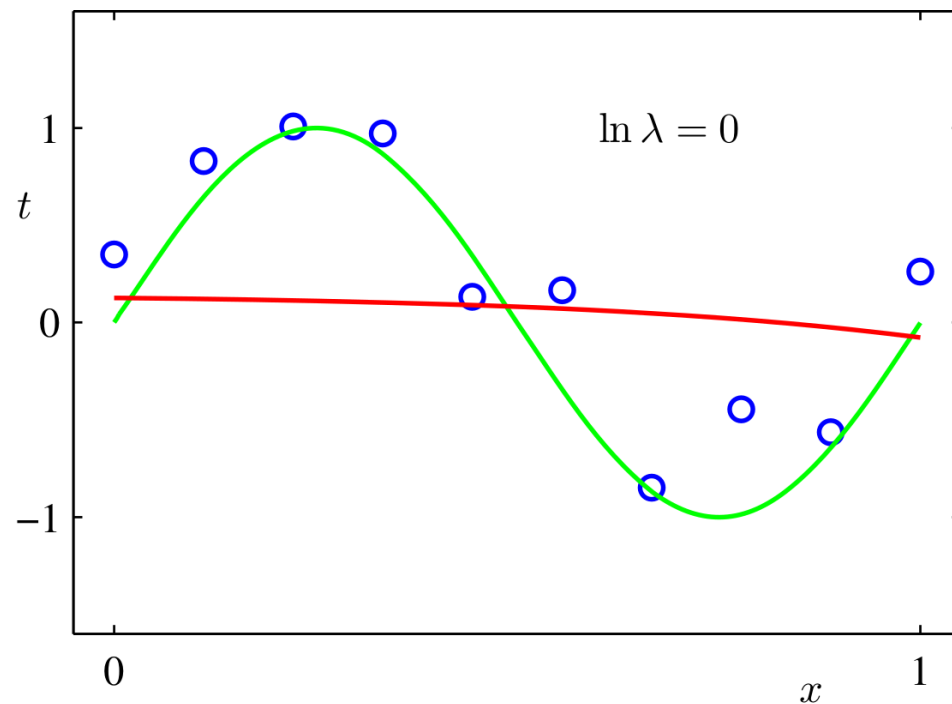
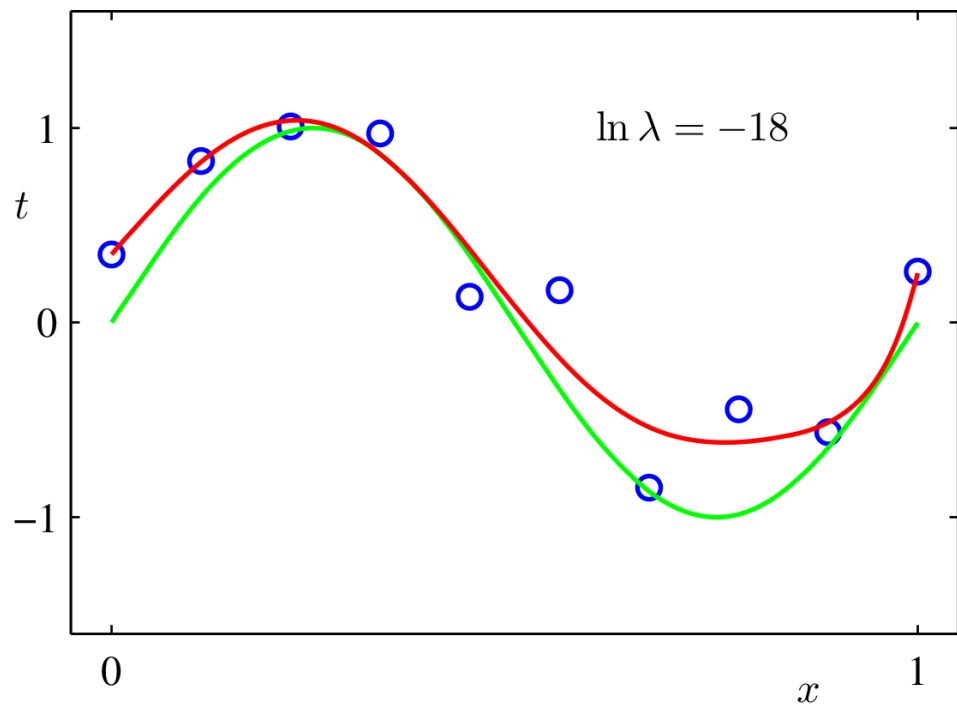


Table 1.2 Table of the coefficients w^* for $M = 9$ polynomials with various values for the regularization parameter λ . Note that $\ln \lambda = -\infty$ corresponds to a model with no regularization, i.e., to the graph at the bottom right in Figure 1.4. We see that, as the value of λ increases, the typical magnitude of the coefficients gets smaller.

	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
w_0^*	0.35	0.35	0.13
w_1^*	232.37	4.74	-0.05
w_2^*	-5321.83	-0.77	-0.06
w_3^*	48568.31	-31.97	-0.05
w_4^*	-231639.30	-3.89	-0.03
w_5^*	640042.26	55.28	-0.02
w_6^*	-1061800.52	41.32	-0.01
w_7^*	1042400.18	-45.95	-0.00
w_8^*	-557682.99	-91.53	0.00
w_9^*	125201.43	72.68	0.01

Why does it help?

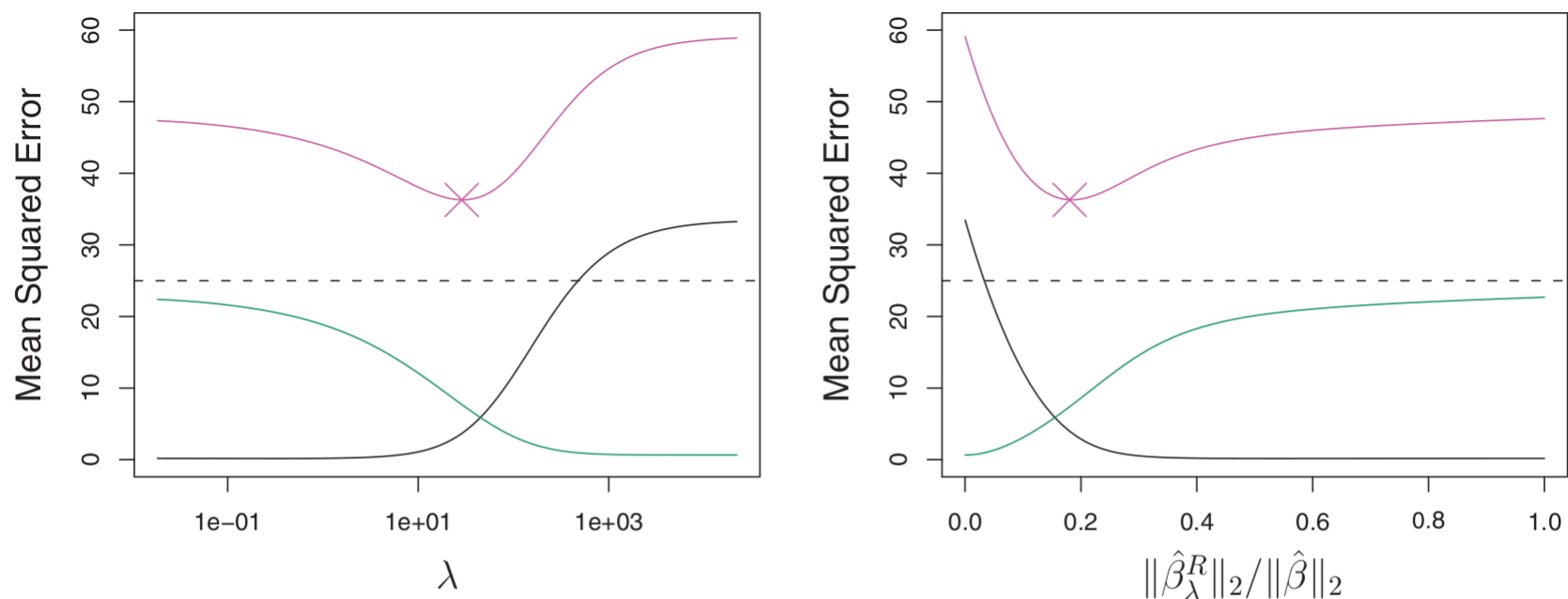


FIGURE 6.5. Squared bias (black), variance (green), and test mean squared error (purple) for the ridge regression predictions on a simulated data set, as a function of λ and $\|\hat{\beta}_\lambda^R\|_2 / \|\hat{\beta}\|_2$. The horizontal dashed lines indicate the minimum possible MSE. The purple crosses indicate the ridge regression models for which the MSE is smallest.

Brief aside on normalization

- Ridge regression is not *scale invariant*
 - Multiply every feature by k, the solution is not simply multiplied by 1/k
 - Least-squares **is** *scale invariant*
- Normalization: Shifting, scaling data before feeding into the model
- Specifically, **standardization**
 - Make each feature zero mean, unit variance by
 - subtracting the mean
 - dividing by standard deviation

$$x \leftarrow \frac{x - \bar{x}}{\text{sd}(x)}$$

- **NOTE** never peak at the test data
 - Calculate mean and sd on the training data, NOT the full dataset

l_1 regularization

- Known as Lasso, Basis Pursuit
- Use l_1 instead of l_2
- Solve

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

- No more closed form solution
- But efficient algorithms available
 - Convex optimization problem

What does l_1 regularization do?

- Pushes β towards zero
- Sets some to exactly zero
 - Results in sparse solution
 - Automatic variable (feature) selection

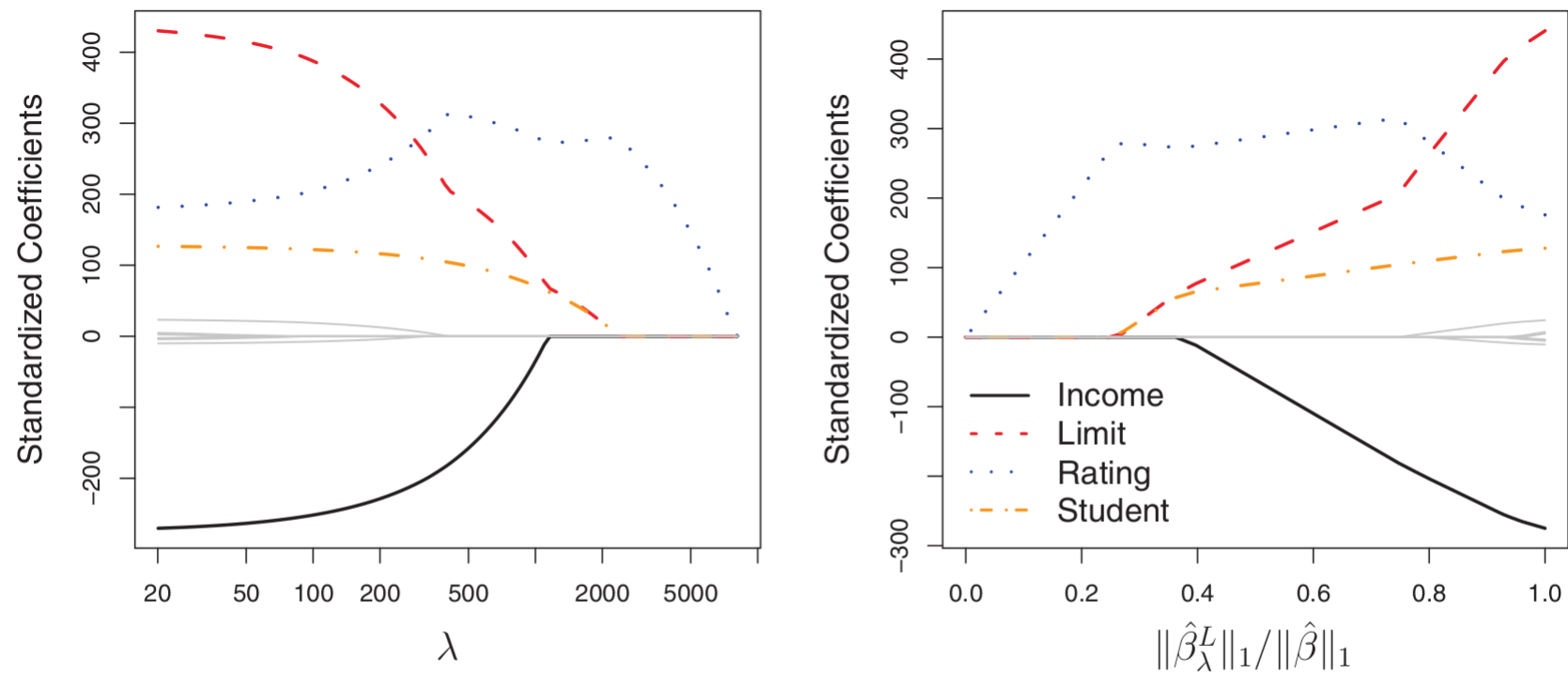


FIGURE 6.6. The standardized lasso coefficients on the **Credit** data set are shown as a function of λ and $\|\hat{\beta}_\lambda^L\|_1 / \|\hat{\beta}\|_1$.

Why does l_1 lead to sparse solutions?

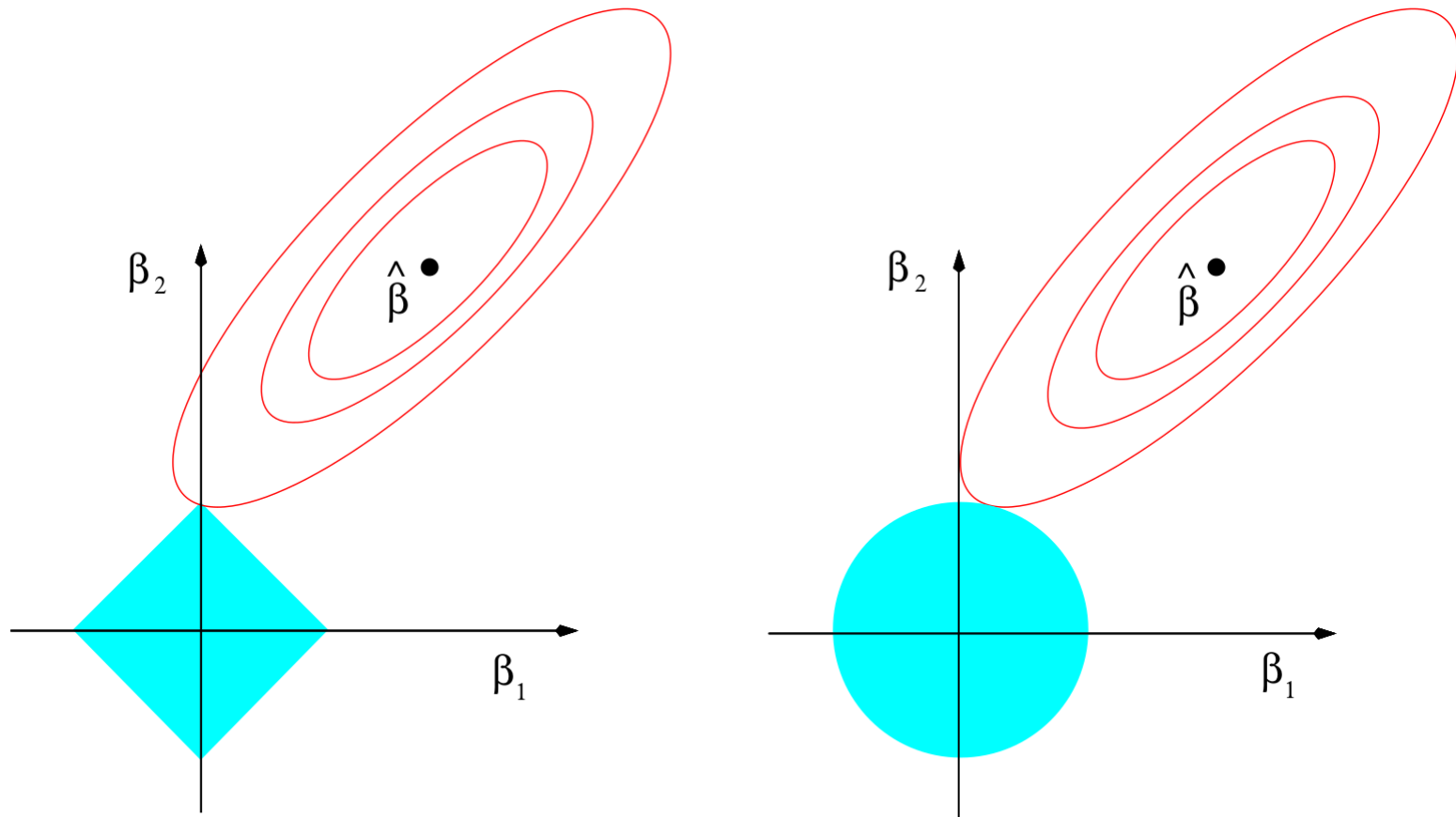


FIGURE 3.11. Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions $|\beta_1| + |\beta_2| \leq t$ and $\beta_1^2 + \beta_2^2 \leq t^2$, respectively, while the red ellipses are the contours of the least squares error function.

Ridge (l_2) vs. Lasso (l_1)

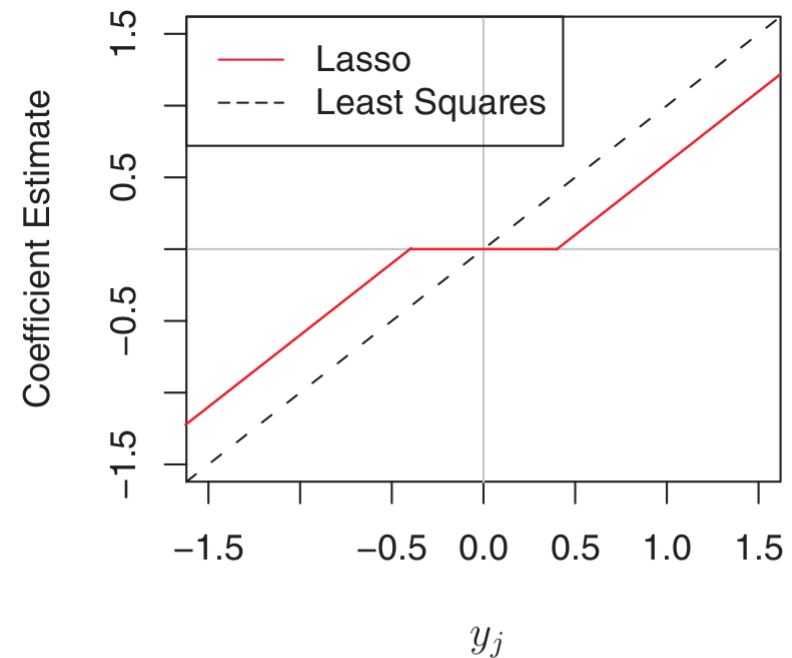
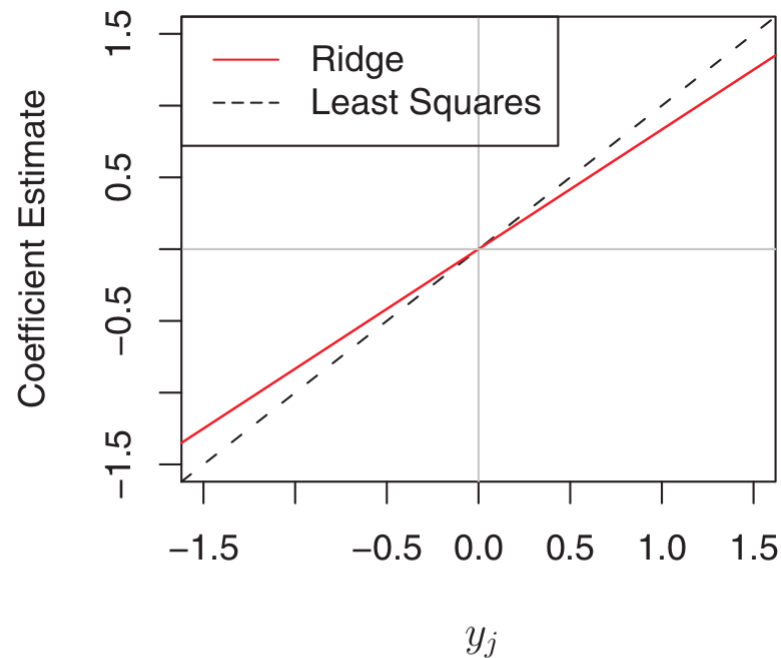
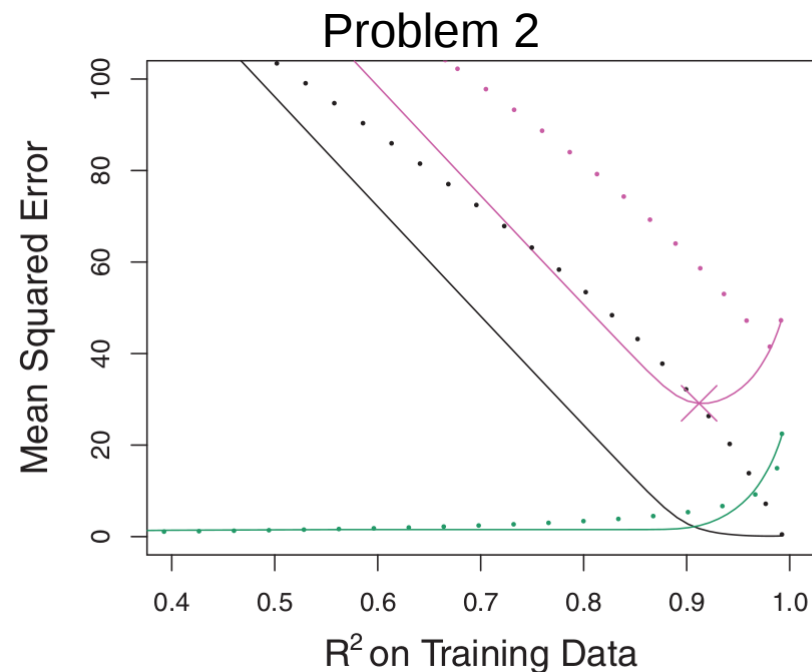
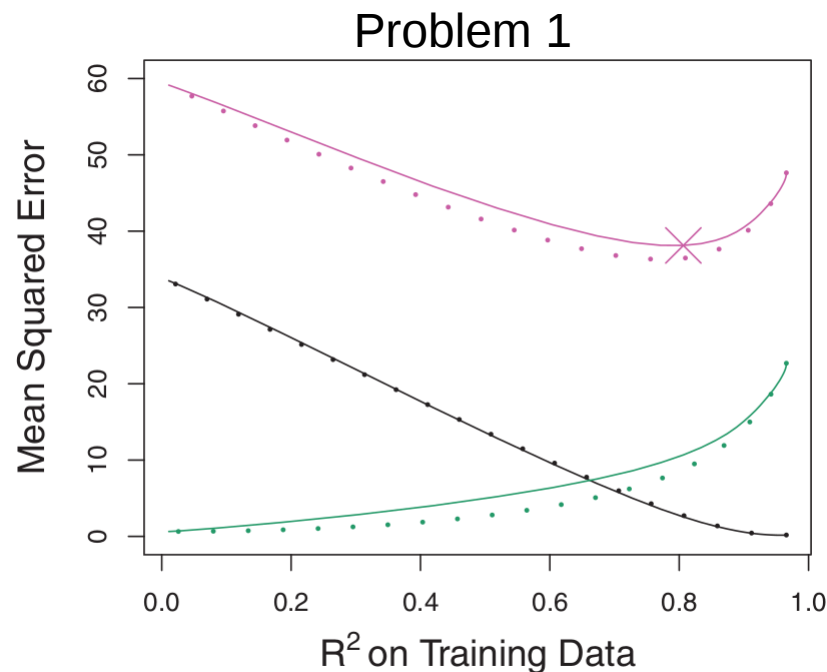


FIGURE 6.10. *The ridge regression and lasso coefficient estimates for a simple setting with $n = p$ and \mathbf{X} a diagonal matrix with 1's on the diagonal. Left: The ridge regression coefficient estimates are shrunk proportionally towards zero, relative to the least squares estimates. Right: The lasso coefficient estimates are soft-thresholded towards zero.*

Ridge (l_2) vs. Lasso (l_1)

- Which one is better? Depends on the problem.
 - If some variables are not related to output at all, lasso might be better

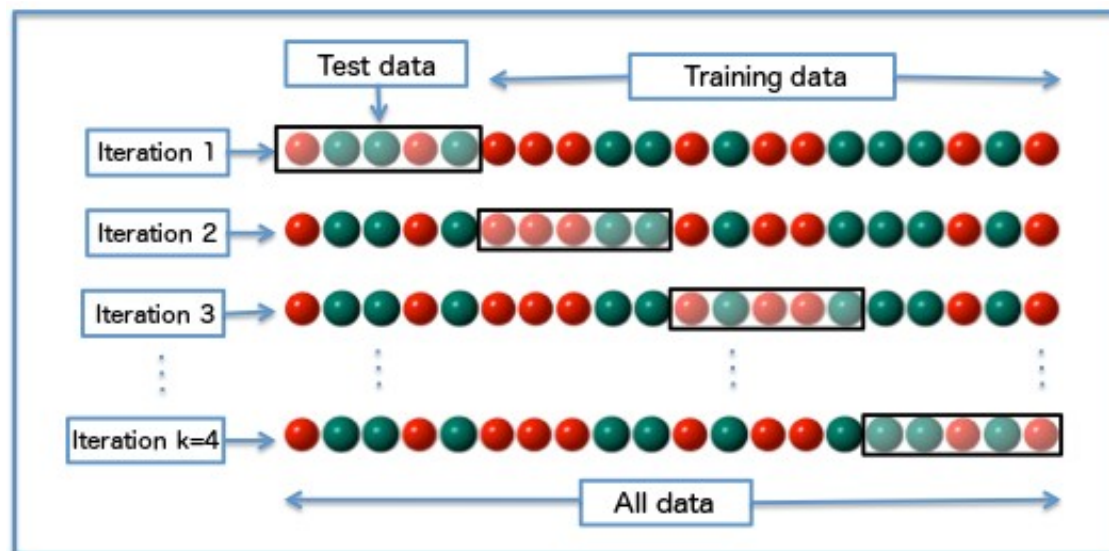


Legend:

Bias (black), variance (green), test error (purple)
Lasso (solid), Ridge (dotted)

Cross-validation

- Estimating the test-performance for a model
 - Pick hyper-parameters (e.g., λ)
- We talked about simple validation (**holdout** method)
 - Split into train/test, evaluate on test
 - Gives you a point estimate
- Various cross-validation techniques
- K-fold cross-validation



Summary

- Regularization
- l_2 regularization
 - Non-sparse solutions
- l_1 regularization
 - Sparse solutions
- Picking λ
 - Cross-validation
- Exercises
 - Derive ridge regression solution
 - Do the lab in Section 6.6 of ISLR

References

- [1] James, Witten, Hastie, and Tibshirani. An Introduction to Statistical Learning with Applications in R. Chapter 6.
- [2] Hastie, Tibshirani, and Friedman. The Elements of Statistical Learning. Chapter 3.
- [3] <https://www.datasciencecentral.com/profiles/blogs/intuition-behind-bias-variance-trade-off-lasso-and-ridge>
- [4] [https://en.wikipedia.org/wiki/Cross-validation_\(statistics\)](https://en.wikipedia.org/wiki/Cross-validation_(statistics))
- [5] https://en.wikipedia.org/wiki/Taxicab_geometry
- [6] [https://en.wikipedia.org/wiki/Norm_\(mathematics\)#Properties](https://en.wikipedia.org/wiki/Norm_(mathematics)#Properties)