Universidade Federal do Paraná

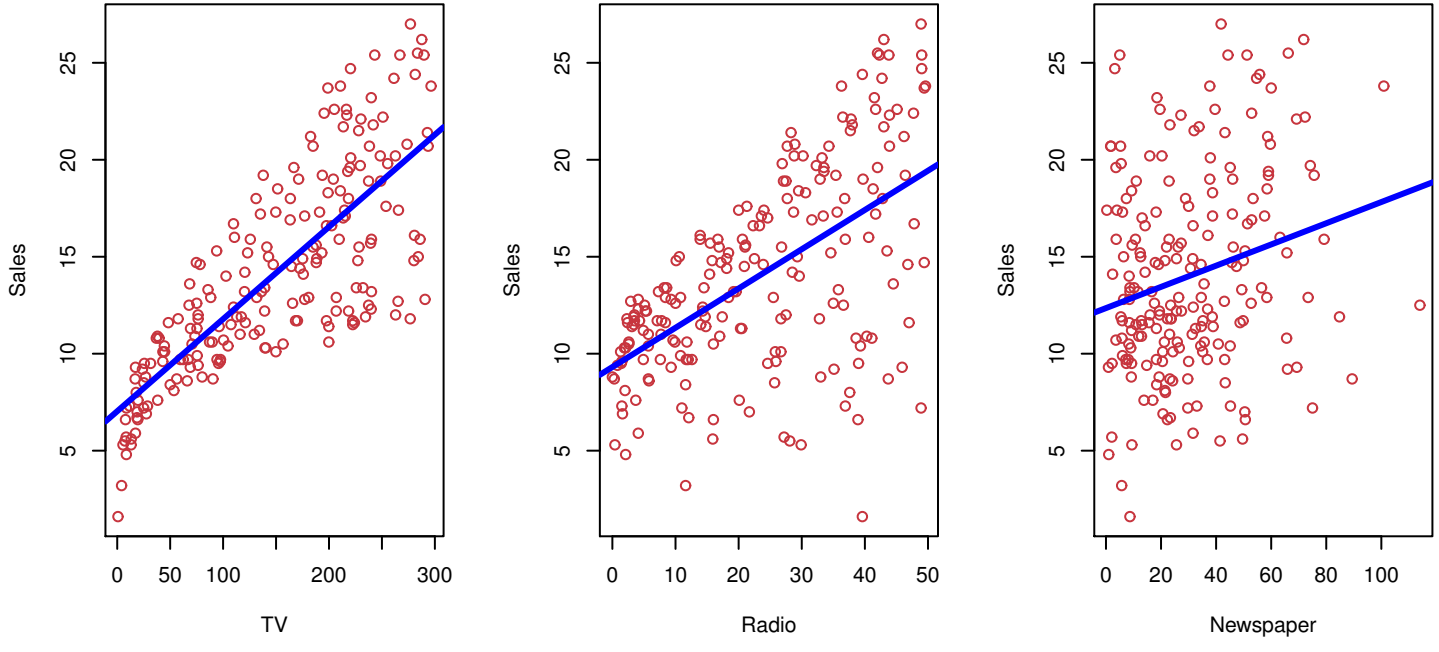
Laboratório de Estatística e Geoinformação - LEG

Regularização

Profs.: Eduardo Vargas Ferreira
Walmes Marques Zeviani

Introdução: Regressão Linear

- Considere o exemplo referente ao `Advertising` data set;
  - ⋆ Y = `Sales` de um particular produto em 200 lojas;
  - ⋆ X = investimento em publicidade na `TV`, `Radio` e `Newspaper` de cada loja.

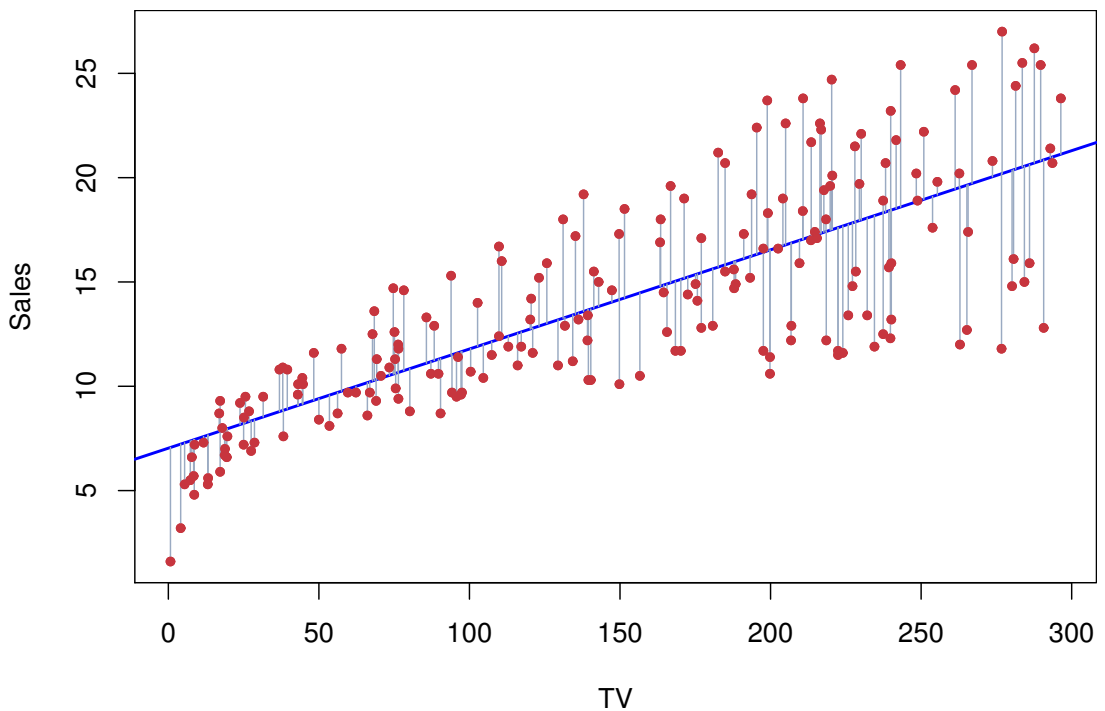


Introdução: Regressão Linear

- Podemos, p. ex., descrever a relação entre `TV` e `Sales` da forma,

$$\texttt{Sales} \approx \beta_0 + \beta_1 \times \texttt{TV}$$

- Queremos encontrar os valores de $\hat{\beta}_0$ e $\hat{\beta}_1$, tq, a reta resultante seja a mais próxima possível dos pontos;



Estimando os coeficientes

- Existem várias formas de medir a proximidade;
- A mais comum envolve minimizar o **Residual Sum of Squares (RSS)**

$$
\begin{aligned}
min\{J(y_i, h(\boldsymbol{x}))\} &\approx min\left\{\sum_{i=1}^{n}[y_i - h(x_i)]^2\right\} \\
&= min\left\{\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2\right\}
\end{aligned}
$$

- Assim, o parâmetro estimado é obtido da forma

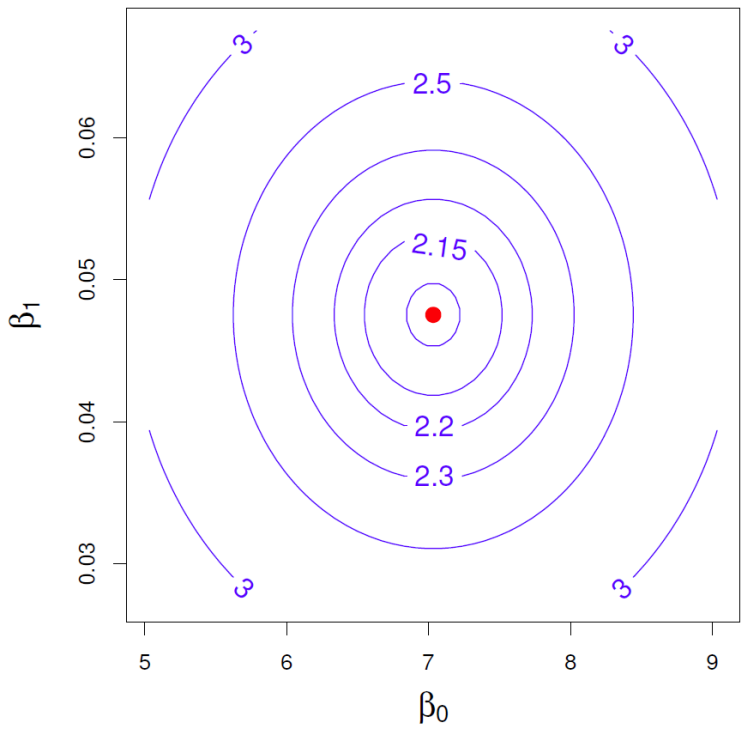
$$\frac{\partial J(y_i, h(\boldsymbol{x}))}{\partial \beta_j} = 0$$

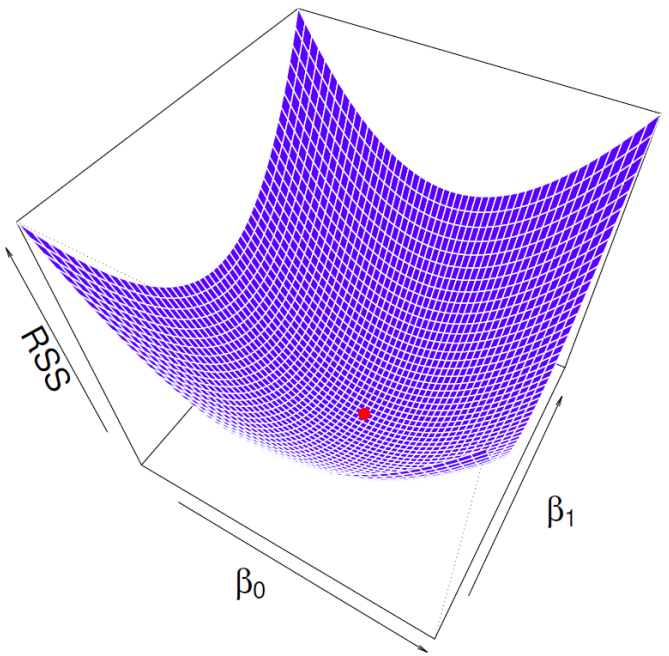
- Chegando em

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^t\boldsymbol{X})^{-1}\boldsymbol{X}\boldsymbol{y}$$

Estimando os coeficientes

- Abaixo o gráfico de contorno da **RSS** para os dados `Advertising`;
- O ponto vermelho representa as estimativas dos parâmetros.





O problema "small n, large p"

- Considere o seguinte problema: queremos verificar se uma substância está ou não relacionada com a incidência de uma doença;
- Cada experimento contém cerca de 5000 variáveis de interesse, baseadas na expressão genética, mas temos apenas 250 animais testados;
- A dificuldade é que a maioria dos métodos modernos de análise de dados falha, por diferentes razões, p. ex.:
  - ⋆ **Modelos Lineares Generalizados** falham, pois a matriz do modelo não tem posto completo;
  - ⋆ **Random Forests** falha, pois a probabilidade de selecionar variáveis importantes diminui muito.
  - ⋆ **Análise de Clusters** e métodos baseados em distâncias no plano cartesiano falham devido à "maldição da dimensionalidade".

O problema "small n, large p"

- Uma suposição razoável é que nem todas as variáveis serão boas para explicar a resposta:
  - ⋆ Algumas serão muito boas;
  - ⋆ Outras que não servem para muita coisa;
- Entretanto, ainda dentro das boas preditoras, algumas são correlacionadas e não são "alavancados" por conta disso.
- Podemos resolver essa questão analisando os vetores um a um (geralmente intratável);
- O ideal seria que o próprio algoritmo realizasse essa seleção;
- Em outras palavras, que o algoritmo regulasse a entrada de algumas variáveis (seja eliminando-as ou diminuindo seus pesos).

Uma solução é a Regularização

- Uma forma de restringir o número de variáveis é impor um custo, ou *penalty*, ao algoritmo

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \text{ sujeito a } P(\boldsymbol{\beta}) < t,$$

em que t é um número real entre zero e infinito.

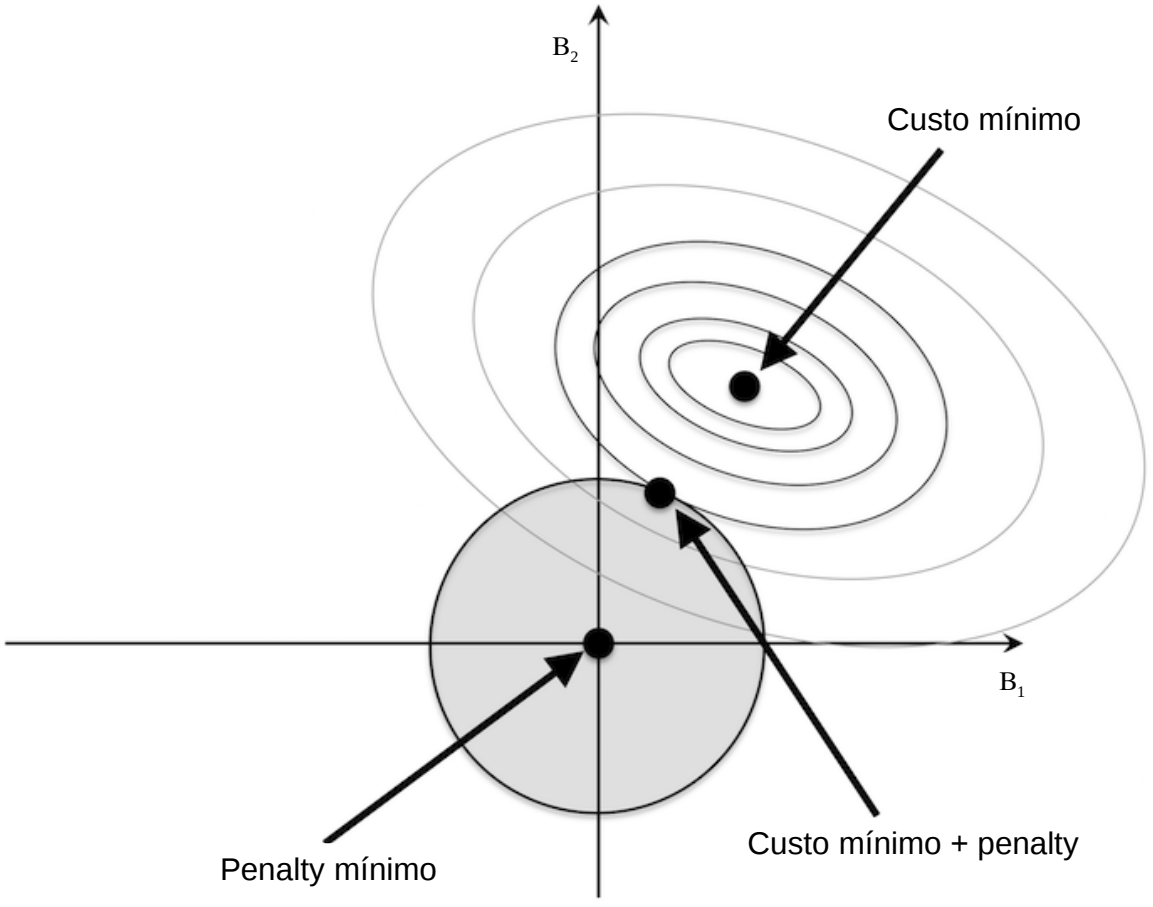
- Fazemos isso aumentando a função objetivo, utilizando os **Multiplicadores de Lagrange**

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

em que λ é um número real entre zero e infinito.

- Note que t e λ são inversamente proporcionais.

Uma solução é a Regularização



Função penalty

- $P(\beta)$ representa a **função penalty** (*shrinkage penalty*), que tem o papel de manter as estimativas de β_j próximas de zero.

- Utilizaremos a **família das potências** para penalizar o modelo, ou seja:

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

- Quanto maior o valor absoluto do coeficiente, maior a penalidade atribuída a ele;

- λ é o **tuning parameter**, determinado separadamente. Ele controla o impacto de J e P nas estimativas dos parâmetros.

$q = 2$ - **Penalização Ridge**

- Neste caso, o problema de otimização é dado por:

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

- A ideia surgiu para solucionar a singularidade da matriz quando $p > n$. Para isso, soma-se uma constante λ à sua diagonal;

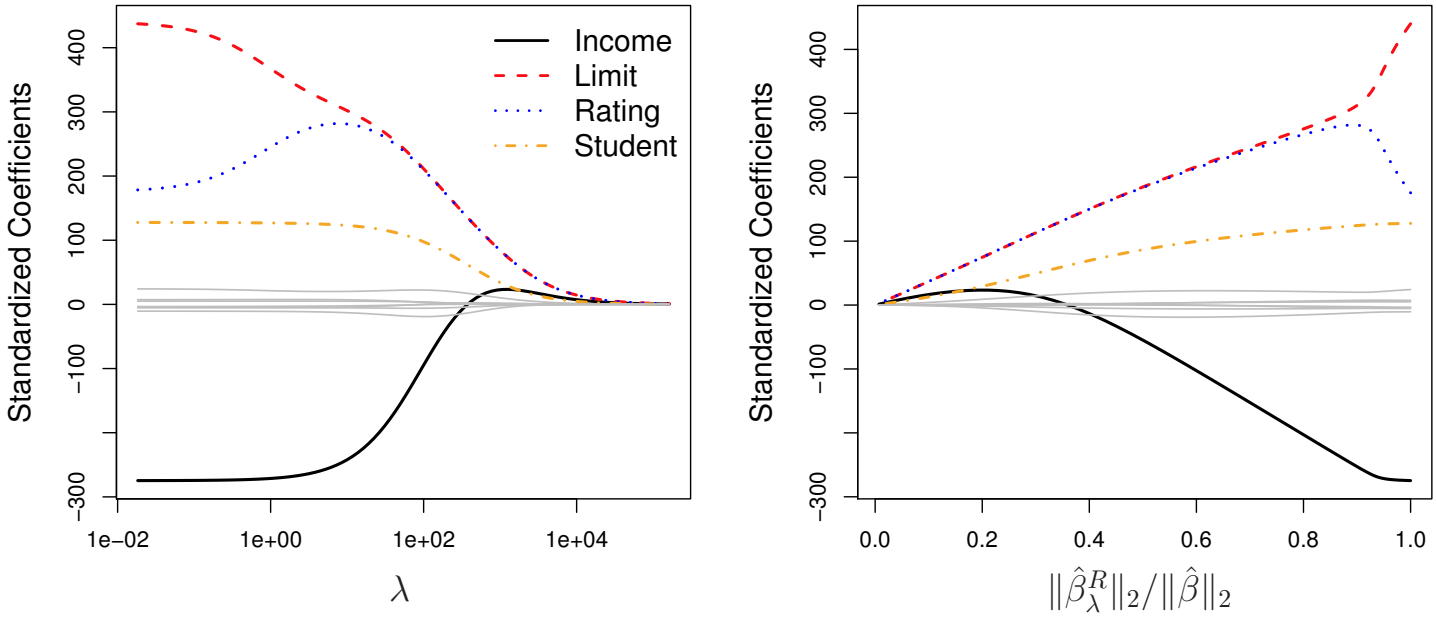
$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T \boldsymbol{X} + \lambda \boldsymbol{I} \right)^{-1} \boldsymbol{X}^T \boldsymbol{y}$$

- Considerando o caso dos vetores de $\boldsymbol{X}$ ortonormais temos

$$\hat{\beta}_\lambda^R = \frac{\hat{\beta}^{OLS}}{1 + \lambda}$$

- Este fato ilustra a característica essencial da regressão *Ridge*: **shrinkage**.

Exemplo: `Credit` data set



- O gráfico da esquerda, cada curva corresponde à estimativa de cada coeficiente através da regressão *Ridge* como função de λ;
- O lado direito refere-se às mesmas estimativas dos coeficientes da regressão, mas como função de $||\hat{\beta}^R_\lambda||_2/||\hat{\beta}||_2$;
- $\hat{\beta}$ denota o vetor de estimativas por mínimos quadrados.

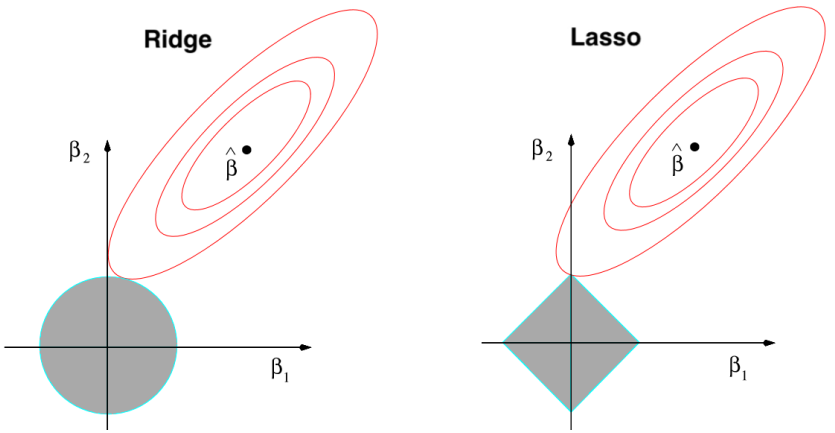
$q = 1$ - **Penalização Lasso**

- A regressão *Ridge* falha na parcimônia do modelo. Ela inclui todos os p preditores (ainda que com pouco peso);

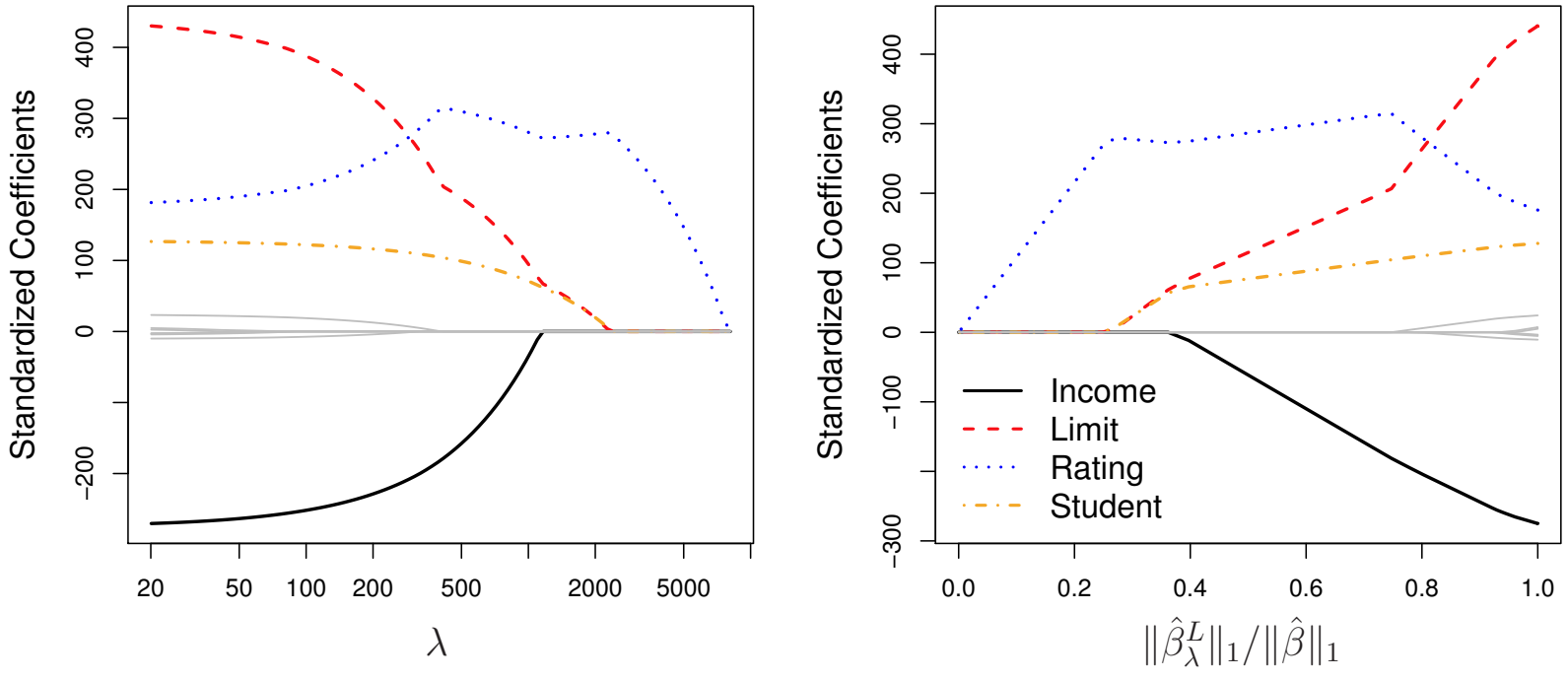
- **Lasso** é uma alternativa que contorna essa desvantagem. Os coeficientes *Lasso*, $\hat{\beta}^L_\lambda$, minimizam a quantidade

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

- O *penalty* ℓ^1 funciona, também, como um **selecionador de variável**.



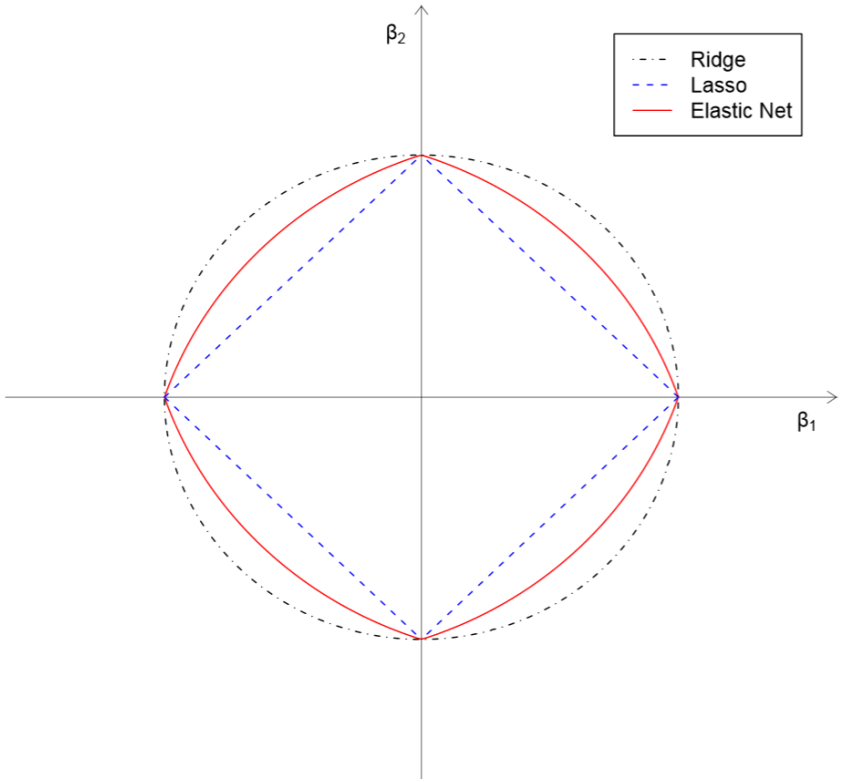
Exemplo: `Credit` data set



Elastic net

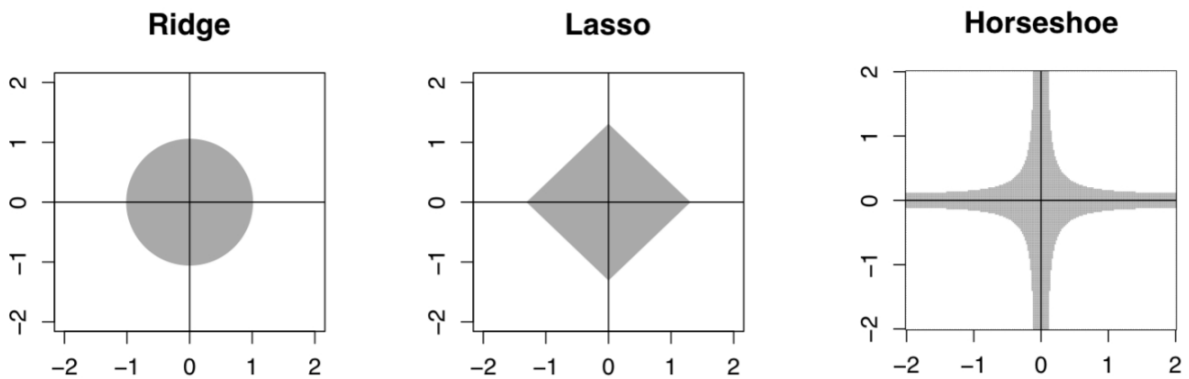
- **Elastic net** é um compromisso entre a regressão *Ridge* e *Lasso*. Os coeficientes elastic net, $\hat{\beta}^E_\lambda$, minimizam a quantidade

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



$q < 1$ - **Penalização horseshoe**

- E o que acontece se reduzirmos q ainda mais? Esse estudo deu origem aos estimadores baseados em penalização *horseshoe*;
- Ela favorece ainda mais a presença de 0's (maior esparsidade);
- Ou seja, tende a encontrar as elipses geradas pelos mínimos quadrados em cima dos eixos com mais frequência que *Ridge* e *Lasso*;



- E quando $q = 0$ voltamos ao **Best subset selection**.

Selecionando o tuning parameter, λ

- Assim como no *subset selection*, para regressão *Ridge* e *Lasso* necessitamos de um método para determinar qual o melhor modelo;
- Neste caso, precisamos encontrar o valor de λ que fornece esta informação;
- **Validação cruzada** fornece uma maneira simples de resolver este problema:

  (a) A partir de uma grade de valores de λ, calculamos a taxa de erro de validação (para cada λ);

  (b) Escolhemos o valor de λ que fornece a menor taxa de erro;

  (c) Ajustamos novamente o modelo, utilizando todas as observações disponíveis, com o valor de λ encontrado anteriormente.

Selecionando o tuning parameter, λ