Regularization and feature engineering

Lecture 10 of “Mathematics and AI”

Outline

1. The curse of dimensionality
2. Regularization

   Gradient descent, ridge regression, lasso
3. Feature engineering and dimension reduction

   PCA, PCR

THE CURSE OF DIMENSIONALITY

The curse of dimensionality

For $p > n$, some features must be collinear.
No unique least-squares solution!

More features → Larger model → Better training accuracy → Worse test accuracy

WANTED

efficient and automatizable way to reduce the number of features

Regularization

Gradient descent

- Find a local minimum (or maximum) of a function $f(x)$ through an iterative search guided by $\nabla f(x)$

$$x_{t+1} = x_t - \gamma \nabla f(x_t)$$

Find local minimum of f

Step size

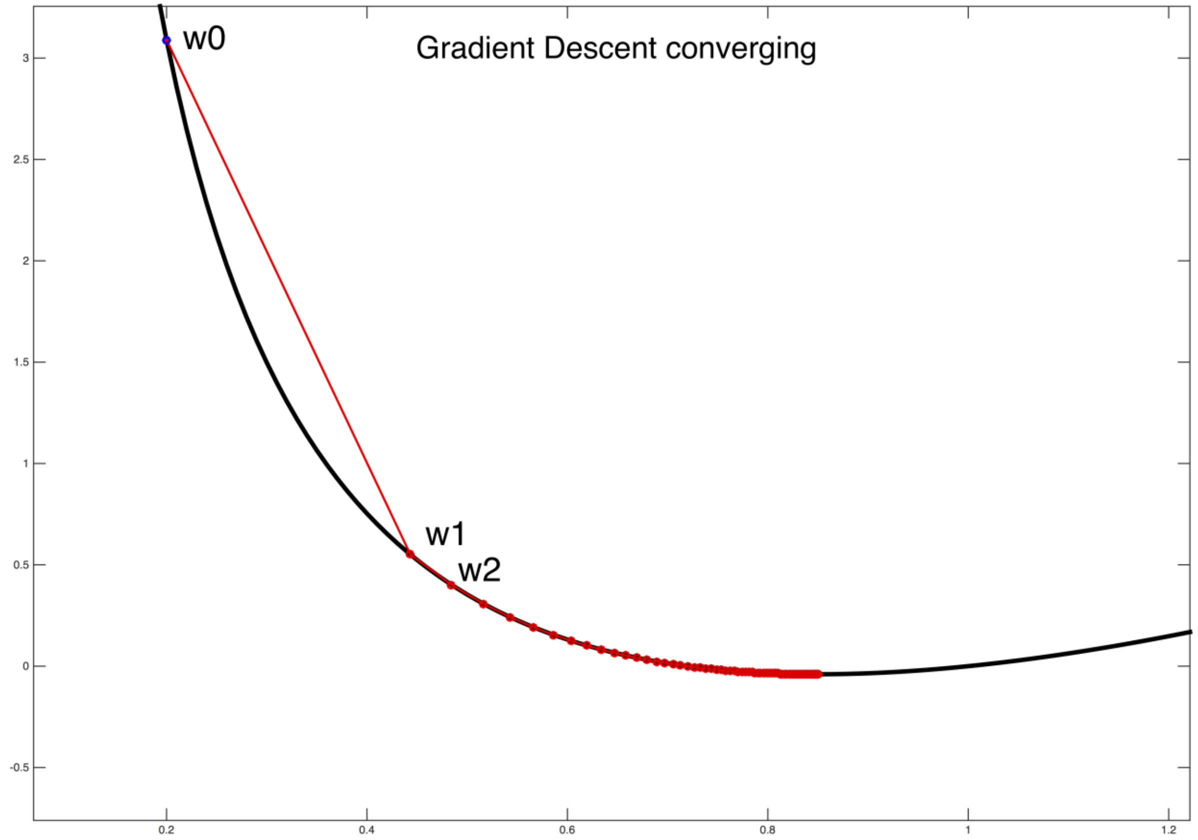
$$x_{t+1} = x_t + \gamma \nabla f(x_t)$$

Find local maximum of f

Step size

- Best use case: Smooth surfaces with few local extrema and saddle points

Convergence and step size



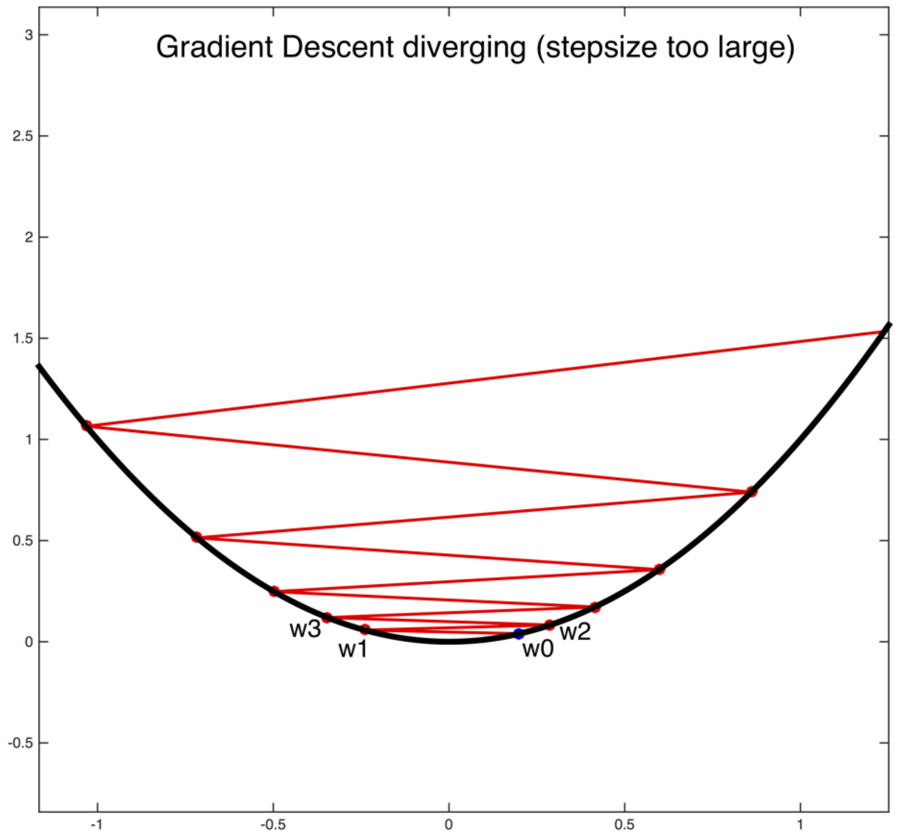


Image source: https://www.cs.cornell.edu/courses/cs4780/2017sp/lectures/lecturenote07.html

Regularization

- Objective function of ordinary least squares (OLS)

$$\min_{\beta_0,\beta_1,\ldots}(\text{RSS}) = \min_{\beta_0,\beta_1,\ldots}\left[\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2\right]$$

- Ordinary least squares with a **regularization** term leads to **coefficient shrinkage**

$$\min_{\beta_0,\beta_1,\ldots}\left(\text{RSS} + \lambda\sum_{j=1}^{p} I[\beta_j \neq 0]\right)$$

best subset selection

$$\min_{\beta_0,\beta_1,\ldots}\left(\text{RSS} + \lambda\sum_{j=1}^{p}|\beta_j|\right)$$

lasso

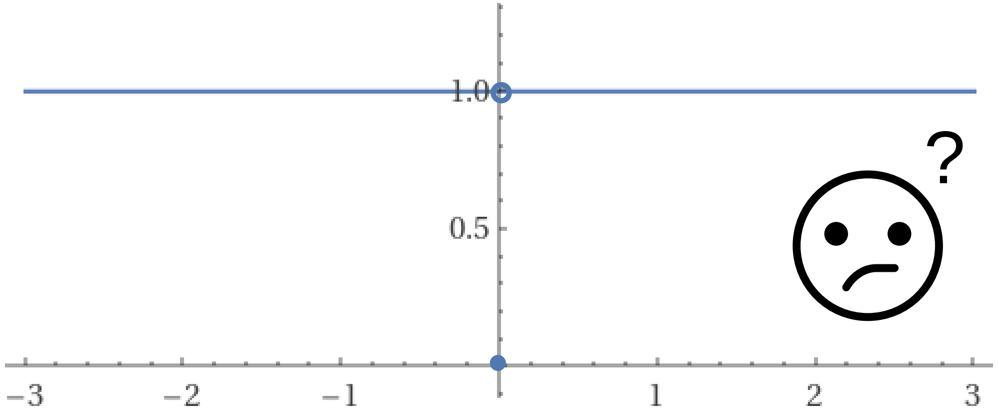
$$\min_{\beta_0,\beta_1,\ldots}\left(\text{RSS} + \lambda\sum_{j=1}^{p}{\beta_j}^2\right)$$

ridge regression

Optimization via gradient descent

best subset selection

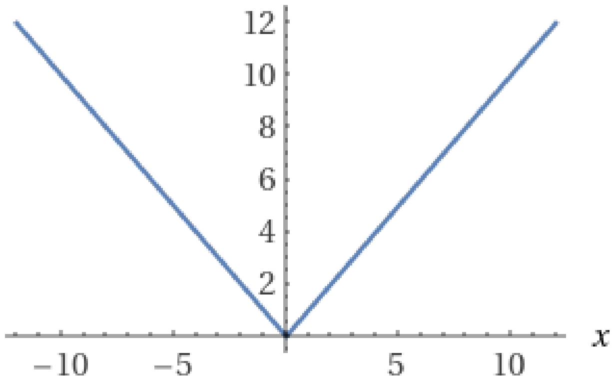
$$\min_{\beta_0,\beta_1,\ldots}\left(\text{RSS}+\lambda\sum_{j=1}^{p} I[\beta_j \neq 0]\right)$$



Computed by Wolfram|Alpha

lasso

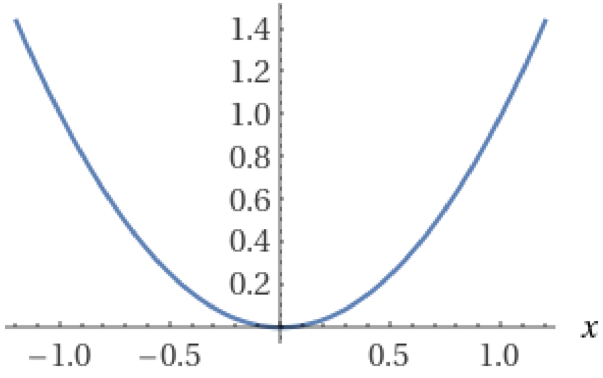
$$\min_{\beta_0,\beta_1,\ldots}\left(\text{RSS}+\lambda\sum_{j=1}^{p} |\beta_j|\right)$$



Computed by Wolfram|Alpha

ridge regression

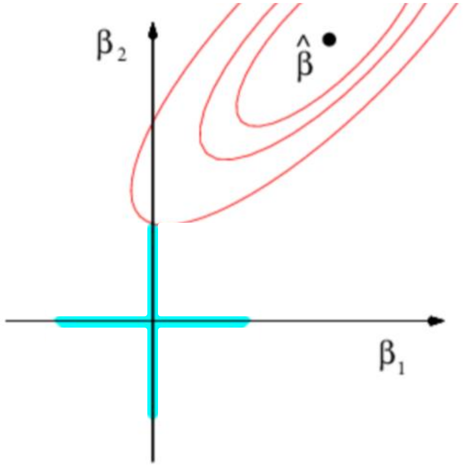
$$\min_{\beta_0,\beta_1,\ldots}\left(\text{RSS}+\lambda\sum_{j=1}^{p} {\beta_j}^2\right)$$

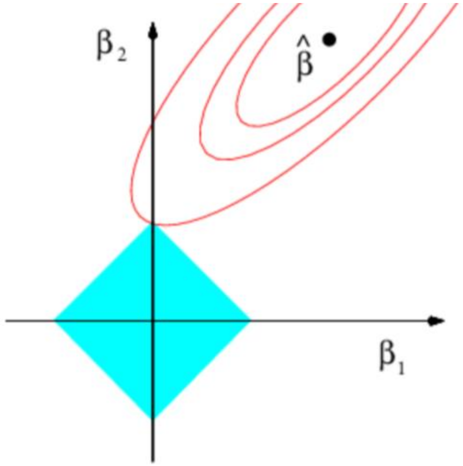


Computed by Wolfram|Alpha

Regularization as optimization constraints

Expression of regularized regression method	Best subset selection	Lasso	Ridge regression
Expression with tuning parameter λ	$\min_{\beta_0,\beta_1,\ldots}\left(\mathrm{RSS}+\lambda\sum_{j=1}^{p} I[\beta_j \neq 0]\right)$	$\min_{\beta_0,\beta_1,\ldots}\left(\mathrm{RSS}+\lambda\sum_{j=1}^{p} \lvert\beta_j\rvert\right)$	$\min_{\beta_0,\beta_1,\ldots}\left(\mathrm{RSS}+\lambda\sum_{j=1}^{p} {\beta_j}^2\right)$
Expression with optimization constraint s	$\min_{\beta_0,\beta_1,\ldots}(\mathrm{RSS})$ subject t $\sum_{j=1}^{p} I[\beta_j \neq 0] \leq s$	$\min_{\beta_0,\beta_1,\ldots}(\mathrm{RSS})$ subject to $\sum_{j=1}^{p} \lvert\beta_j\rvert \leq s$	$\min_{\beta_0,\beta_1,\ldots}(\mathrm{RSS})$ subject to $\sum_{j=1}^{p} {\beta_j}^2 \leq s$





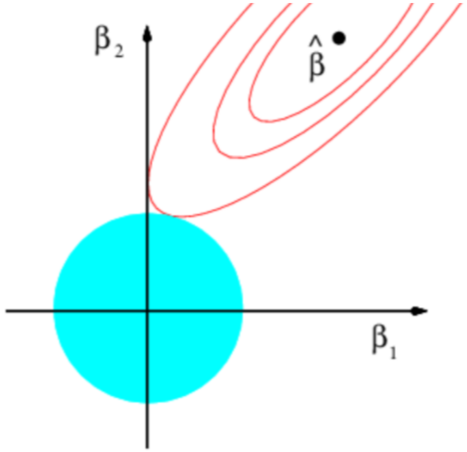
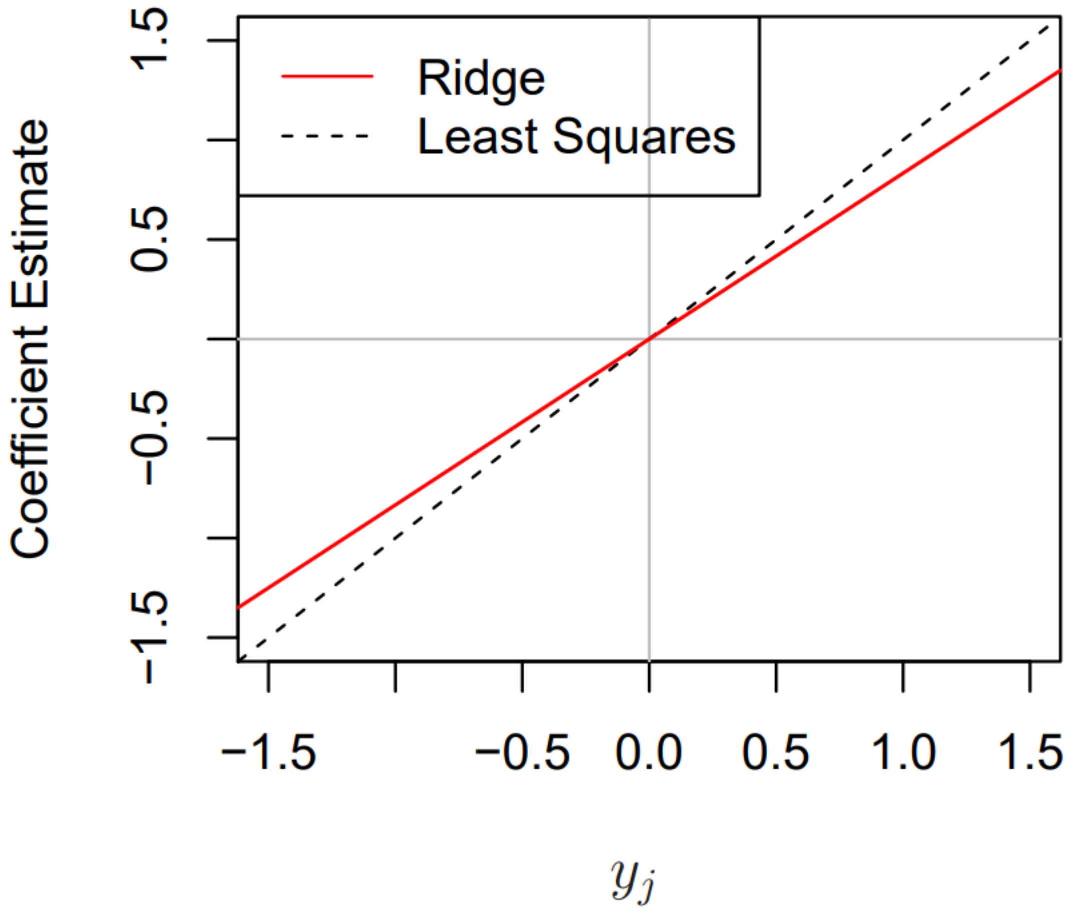


Image source: Introduction to Statistical Learning in Python

Comparison of ridge regression and lasso



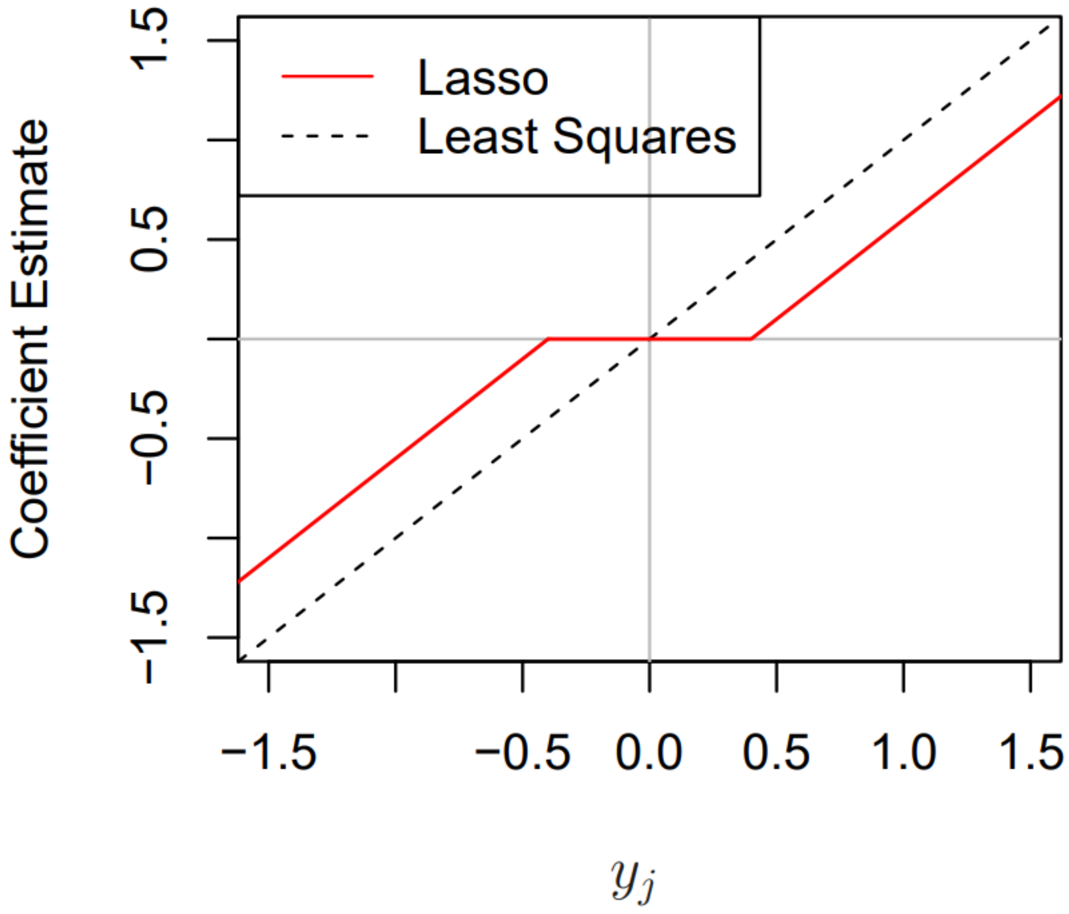


Image source: Introduction to Statistical Learning in Python

Feature engineering / Dimension reduction

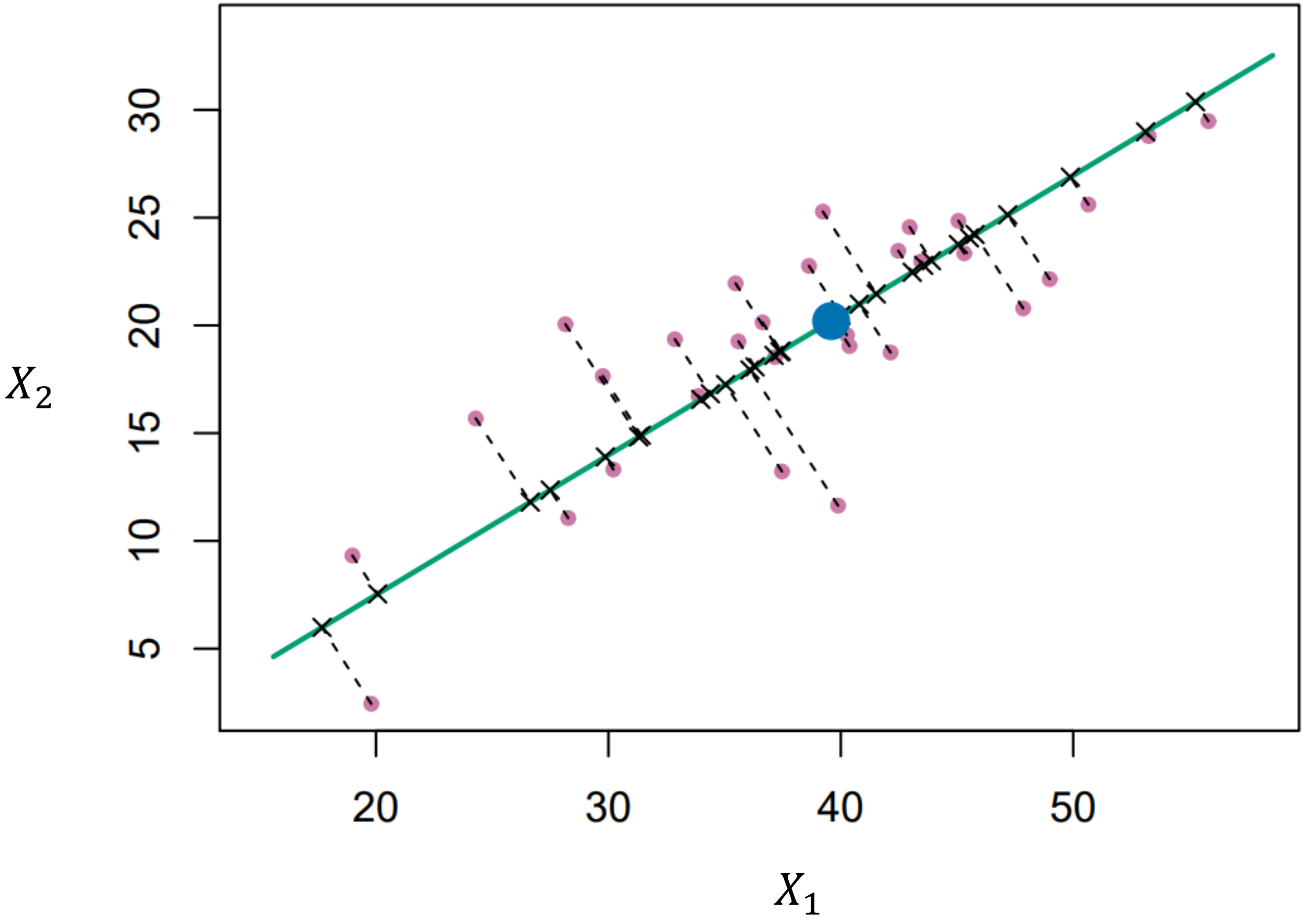
Feature engineering and dimension reduction

- When $p > n$, we want to find construct a set of new m features with $m < n, p$

Feature engineering **Dimension reduction**

- How should we construct new features?
  - Singular value decomposition of input data → Principal component regression
  - Residuals of linear regression → Partial least squares

Principal component analysis (PCA)

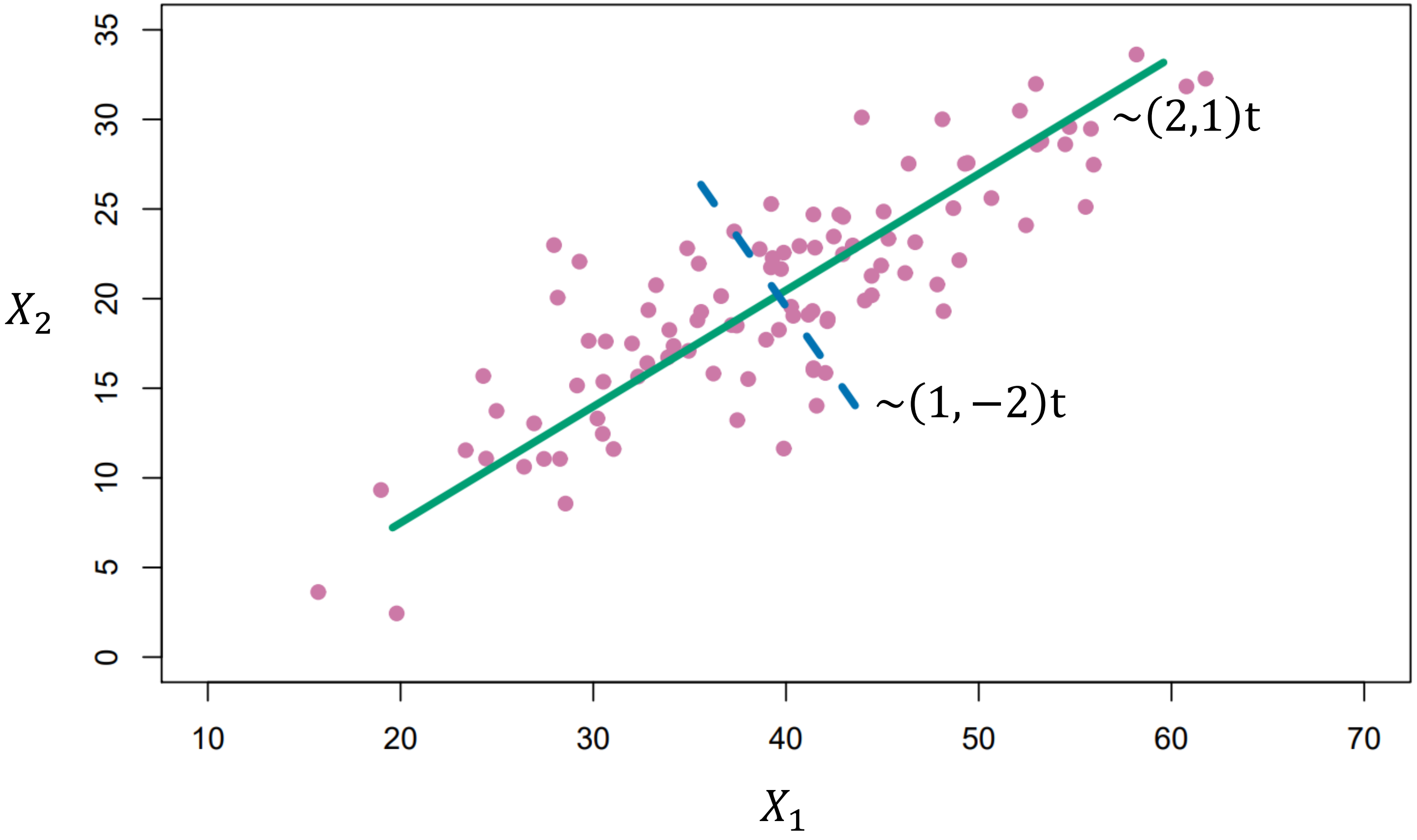


Principal component 1:

Find the line in feature space s.t. the deviations from observations (shown as dotted lines) are minimal.

Image source: Introduction to Statistical Learning in Python

Principal component analysis (PCA)



Principal component 1
$z_1 = 2x_1 + x_2$

Principal component 2
$z_2 = x_1 - 2x_2$

Principal components are the directions of greatest variation!

Singular vectors of mean-centered data!

Image source: Introduction to Statistical Learning in Python

Principal component analysis (PCA)

- General setting:

$$Z_k = \sum_{j=1}^{p} \varphi_{jk} X_j \qquad \text{for } k = 1, \dots, m$$

Loading of variable X_j on principal component Z_k

- Principal component regression
  - ➢Use the first m principal components Z_k as new features
  - ➢ Imposes constraints on regression coefficients compared to OLS on X_j