Group 3: Regression

1. Introduction

- Our group focuses on **linear regression** problems
- Data and method selection based on The Elements of Statistical Learning
- The goal of a regression is to inspect a possible dependency of Y given X
    - $Y \in \mathbb{R}$ dependent variable
    - $X \in \mathbb{R}^p$ independent variable
        - Where an instance is a vector x containing p measurements

1. Introduction

Basic idea behind any regression

Optimize the following problem:

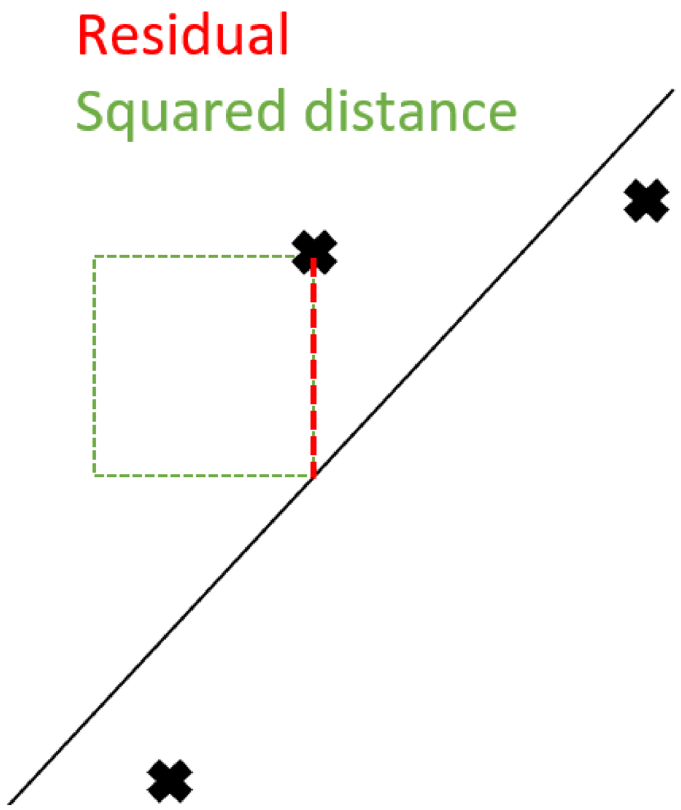
- $\underset{f}{argmin}\ L(Y - f(X))$

Example - Linear Regression

- Minimize $\epsilon^t \epsilon = (y - X\beta)^t (y - X\beta)$
- $\underset{\beta}{argmin}\ (y - X\beta)^t (y - X\beta)$

Leads to the Ordinary Least Square Estimator:

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



2. Our Datasets

- UCI Data Repository
  - real_estate
  - winequality-red
  - AirQualityUCI
- Data proposed in Elements of Statistical Learning
  - Prostate cancer

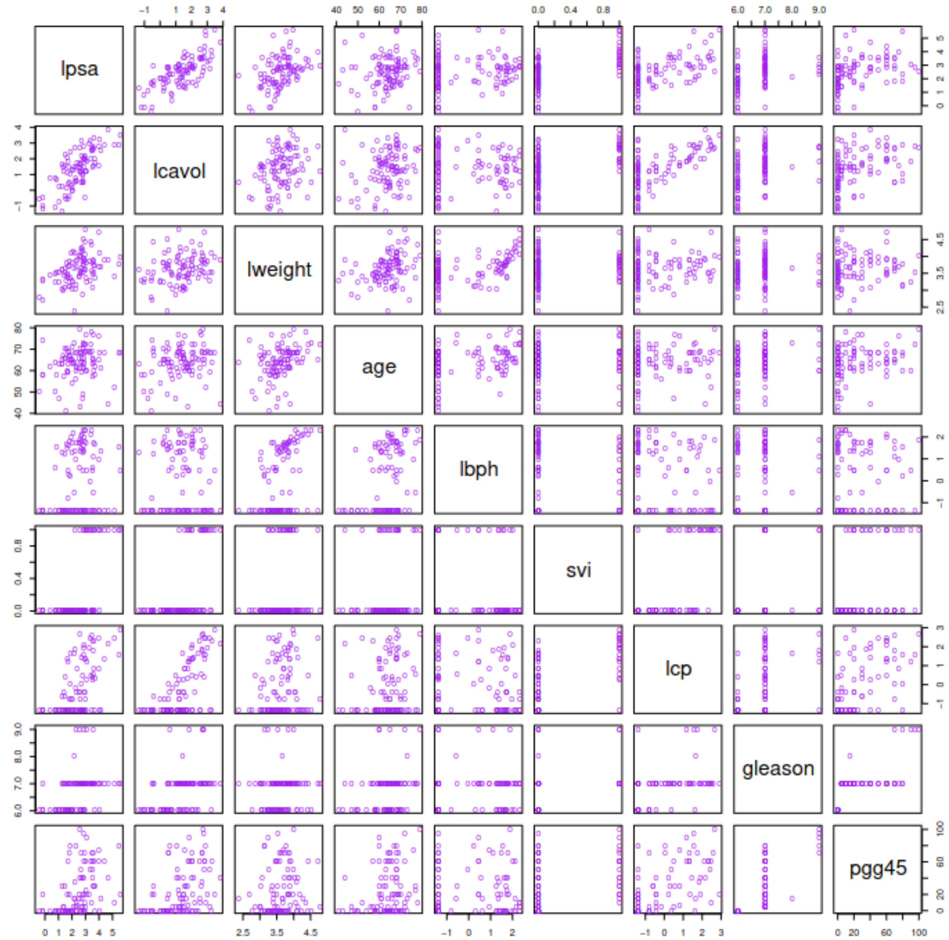
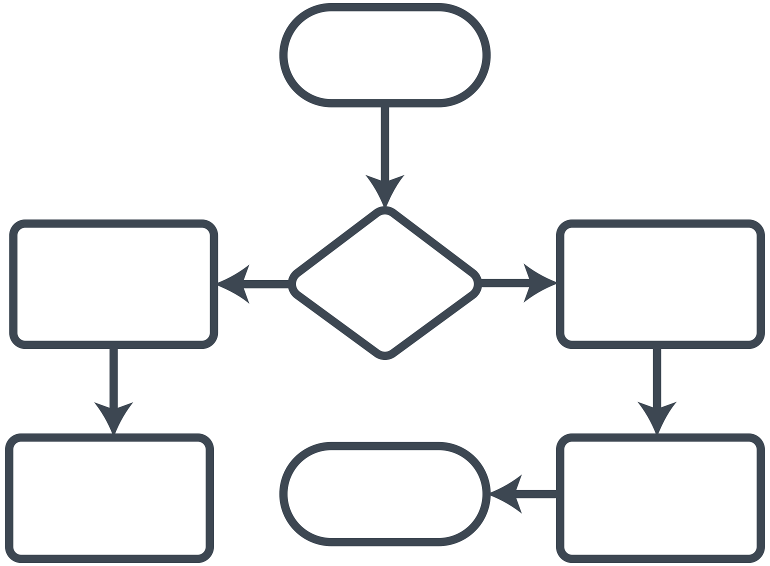


FIGURE 1.1. *Scatterplot matrix of the prostate cancer data. The first row shows the response against each of the predictors in turn. Two of the predictors,* `svi` *and* `gleason`*, are categorical.*

3. Comparison of Regression Models

Workflow

1. Implement regression methods using python libraries
2. Implement selected models from scratch
3. Evaluate and compare the implemented models
    - Metric-performance (with default parameters)
    - Speed-performance
    - Memory-performance



3. Comparison of Regression Models

i. Ordinary Least Squared

Idea: find best-fitting line $f(x) = m + xb$ for Y given X.

$\hat{\beta} = (X^tX)^{-1}X^ty = (X^tX)^{-1}X^t(X\beta + \epsilon)$

- Good baseline with zero estimation bias
  $\mathbb{E}[\hat{\beta}] = \cdot \cdot_{math-magic}$🧙$\cdot \cdot \quad = \beta + (X^tX)^{-1}X^t\mathbb{E}[\epsilon]$
  since $\mathbb{E}[\epsilon] = 0$ since $\epsilon \sim N(0, \sigma^2)$ it follows $\mathbb{E}[\hat{\beta}] = \beta$
- Simple linear Algebra
- More complex methods trade off for benefits of reduced variance

3. Comparison of Regression Models

ii. Elastic Net (Lasso & Ridge)

Idea: tune down dimensions of X that have little to no influence on Y.

Approach: Regularize Estimator $\hat{\beta}$ with respect to $|\hat{\beta}|$

- $\underset{\beta}{argmin}\, ||y - X\beta||^2 + \lambda_1||\beta||^2 + \lambda_2||\beta||_1$
- Elastic Net extends Ordinary Least Squares
- **Lasso** adds a penalty based on the l_1-norm of the coefficients
- **Ridge** adds a penalty based on the l_2-norm of the coefficients
- Choice of λ_1 and λ_2 are additional constraints for optimization problem

3. Comparison of Regression Models

iii. Least Angle Regression

- **LAR** is a relative newcomer *(Efron et al., 2004)*
- "Democratic" version of forward stepwise regression
- Extremely efficient algorithm for computing the entire lasso path ($\lambda_1 \rightarrow \infty$ until convergence).

3. Comparison of Regression Models

iv. Locally Weighted Regression (LWR)

Idea: Ordinary Least Squares but now certain data points get more weight than others

Approach: Construct weights (Matrix W)

- $\mathcal{L}_{weighted}(\beta) = (y - X\beta)^T \, W(y - X\beta)$

Locally weighted regression:

- Locally put emphasis on points in low proximity
- In total E independent weighted regressions
- $\mathcal{L}_{weighted}(\beta) = (y - X\beta)^T \, W_E(y - X\beta)$
- E.g. $w_i = e^{\frac{-(x_i - x)^2}{2\tau^2}}$

3. Comparison of Regression Models

v. Principal Component Regression (PCR)

Idea: Combine PCA and Linear Regression

Approach: Reduce complexity and dimensionality

- Perform PCA to obtain Principal Components (PCs)
- Choose subset of PCs *(explained variance)*
- Regress response on selected PCs treating each as an univariate regression

Key Equations:

- PCA : $Z_m = Xv_m$
- PCR : $\hat{y}^{pcr}_{(M)} = \bar{y}1 + \sum_{m=1}^{M} \hat{\theta}_m z_m$
- Coefficients : $\hat{\beta}^{pcr}(M) = \sum_{m=1}^{M} \hat{\theta}_m v_m$

4. Evaluation

Comparison with standard libraries / procedures

- `scikit-learn`
  - Ordinary Least Squares
  - Elastic Net
  - Lasso LARS
  - PCA Regression
- Public libraries
  - `localreg` (locally weighted regression)

4. Evaluation

For each method and dataset we compute …

- Model Scores:
  - Mean Squared Error
    $\mathrm{MSE}(y, \hat{y}) = \frac{1}{n} \sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2$
  - Mean Absolute Error
    $\mathrm{MAE}(y, \hat{y}) = \frac{1}{n} \sum_{i=0}^{n-1} \left| y_i - \hat{y}_i \right|$
- Performance benchmarks:
  - Runtime
  - Memory

5. Remarks and Outlook

- Usage of "default" settings method hyper-parameters
- Future work: tuning methods for better results
  - By variable / feature selection
  - By engineering new features
  - Estimating hyper-parameters

6. Literature

- Notes on Regularized Least-Squares
- Elements of Statistical Learning (Hastie et al.)
- Many Regression Algorithms, One Unified Model (Stulp, Signaud)
- Scikit-Learn Documentation