Intro to Supervised Learning through the eyes of Linear Regression

Ido Greenberg

2018

Scope

+ Linear regression

+ Major concepts in supervised learning (validation, overfitting, etc.)

- No introduction of any other specific method

Introduction to Supervised Learning through Linear Regression

- What is the **problem**?
- How is it **modeled**?
- **Why** does it make sense?
- **How to solve** it?
- How can I **validate** what I did?
- **Overfitting**
- How can I **do more** with linear regression?

Supervised Learning Problem

- Goal: learn to predict from input $x \in R^m$ some output $y \in R^k$ as generalization from given data of N samples:

$$X = (x_1 \dots x_N)^T \in R^{N \times m} \qquad Y = (y_1 \dots y_N)^T \in R^{N \times k}$$

- Linear Regression model:

$$Y = X\beta + \epsilon$$

  - X is known input, Y is unknown (except for the training data) output
  - $\beta \in R^m$ are the parameters of the model (AKA coefficients) that we need to learn
  - ϵ is unmodeled noise/errors which is often assumed to be Normally-distributed & independent over different data samples

Least Squares Solution

- Find β with "good fit" to the data X,Y under the model
  - Good fit ➔ closer to the "true" model ➔ better prediction of y given new x
- Fit is measured by **Loss (/Cost) function** $\boldsymbol{L_{\beta}(X,Y)}$
  - Often function of the error $L_{\beta} = L(Y - X\beta)$
    - **L_1-loss**: $L = \left|\left|Y - X\beta\right|\right|_1 = \sum |Y_i - X_{i\cdot}\beta|$
    - **L_2-loss**: $L = \left|\left|Y - X\beta\right|\right|_2^2 = \sum (Y_i - X_{i\cdot}\beta)^2$ (popular due to differentiability)
    - **L_{inf}-loss**: $L = \left|\left|Y - X\beta\right|\right|_\infty = \max |Y_i - X_{i\cdot}\beta|$
  - Other example: Y = change of price ➔ Loss = if(sign(Y)=sign(Xβ)) 0 else |Y|
- **Least squares**: optimize L_2-loss ($\underset{\boldsymbol{\beta}}{\mathbf{argmin}}\, ||\boldsymbol{Y} - \boldsymbol{X\beta}||_{\mathbf{2}}$)

Statistical (Bayesian) Justification

$$Y = X\beta + \epsilon$$

- Assume $\epsilon_i \sim N(0, \sigma^2)$ independently over i:

$$P(\epsilon) \propto \Pi_i e^{-\epsilon_i^2/\sigma^2}$$

- Maximum Likelihood:

$$L(\beta|X,Y) \coloneqq P(Y|\beta,X) = P(\epsilon = Y - X\beta)$$

$$\log L(\beta) = -\Sigma_\mathrm{i} \frac{\epsilon_i^2}{\sigma^2} = -\frac{\Sigma_i (Y_i - X_{i\cdot}\beta)^2}{\sigma^2} = -\frac{1}{\sigma^2} \left| |Y - X\beta| \right|_2^2$$

$$\mathbf{argmax}_{\boldsymbol{\beta}}(\mathbf{log}\, \boldsymbol{L}(\boldsymbol{\beta})) = \mathbf{argmin}_{\boldsymbol{\beta}} \left| |\boldsymbol{Y} - \boldsymbol{X\beta}| \right|_{\mathbf{2}}$$

How to Apply Least Squares?

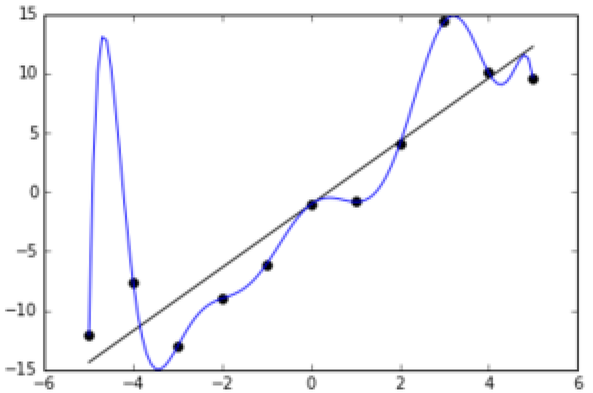
$$\boldsymbol{\hat{\beta}} \coloneqq \left(\boldsymbol{X^T X}\right)^{-1} \boldsymbol{X^T Y}$$

- **Seize the moment** – it may be the last analytically-solved supervised model you'll see for a while.
- "Units" of $\hat{\beta}$ are $[y/x]$ as expected.
- Non-invertible $X^T X$ indicates degenerated X ➔ some variable is a combination of others and can be removed without loss of information.
- Note: $\boldsymbol{X^T X} \in \boldsymbol{R^{m\times m}}$ and $\boldsymbol{X^T Y} \in \boldsymbol{R^{m\times k}}$ are **sufficient statistics of $\boldsymbol{\beta}$ whose size is independent of the number of samples $\boldsymbol{N}$**.

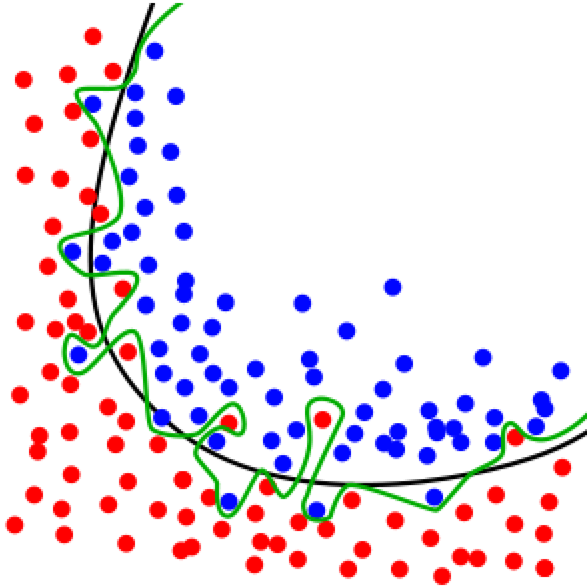
Validation: how can I know that I did well?

- Statistical estimation
  - Assigning range of values for each coefficient
    - Non-significant $\beta \neq 0$ **may** indicate irrelevant input variables
  - Assuming **independent**, normally and identically distributed errors
- Train group vs. test group
  - Cross validation
  - In sequential data: sequential test groups

Overfitting



- Which model is more reasonable?
  - **Occam’s razor**: simplicity should be prioritized
- What is *simple*?
  - Low sensitivity of model to data
    - Less **Degrees of Freedom** (AKA parameters, coefficients)
    - Smaller values of parameters
  - **Bias-variance tradeoff**



- How to reduce variance?
  - Architecture: less parameters (in linear regression – less input variables)
  - **Regularization**: force reduction of β, e.g. by adding $||\beta||$ to the loss function
    - **Lasso** (L_1-penalty), **Ridge** (L_2-penalty)

Getting more from Linear Regression

- Intercept: $\begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ 1 & x_{21} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{pmatrix}$
- Weighted Least Squares
- **Non-linear input**
  - E.g. $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$

Summary
Linear Regression & Supervised Learning

	Linear Regression and Least Squares	Supervised Learning
Problem	Use available data $\{(x_i, y_i)\}_{i=1}^{N}$ ($x_i \in R^m, y_i \in R^k$) to learn to predict y from x in future data $\{x_i\}_i$.	
Model	Linear regression model: $Y = X\beta + noise$	$Y \approx F_\Theta(X)$
Goal	Least squares: $\underset{\beta}{\operatorname{argmin}} \|\|Y - X\beta\|\|_2^2$	$\underset{\Theta}{\operatorname{argmin}}\, L\big(Y, F_\Theta(X)\big)$ L may be defined as $L_1/L_2/L_\infty$ norm of the errors $\|Y - F_\Theta(X)\|$, or as something else.
Bayesian justification	ML (Maximum-Likelihood) for Normal iid noise	Usually as fuzzy as the complexity of the model
Learning	$\hat{\beta} \coloneqq (X^T X)^{-1} X^T y$	Numerical search methods (AKA optimization)
Validation	• Statistical significance (strong assumptions) • Test groups (sequentially / cross validation)	Statistical significance is usually non-practical
Avoid overfitting	• Reduce input size • Penalty for large βs	In complex models, internal model's DOF can also be reduced
Non-linear models	Non-linear input	Built in the model – though input-engineering still tends to be helpful for learning!