PS5841

Data Science in Finance & Insurance

Linear Regression

Yubo Wang

Autumn 2022

Columbia University
in the City of New York

Model

$$E(Y_i) = \mu_i(\boldsymbol{\beta}) = \boldsymbol{x}_i^T \boldsymbol{\beta}$$
$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

- $Y_1, \ldots Y_n$ are independent random variables
- Homoscedasticity

$$Var(Y_i) = \sigma^2, \qquad Var(\boldsymbol{y}) = \sigma^2 I$$
$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_i \left(Y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right)^2$$

- Heteroscedasticity

$$Var(Y_i) = \sigma_i^2, \qquad Var(\boldsymbol{y}) = V$$
$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_i \frac{1}{\sigma_i^2} \left(Y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}\right)^2$$

Least Squares Estimator

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, S_w(\boldsymbol{\beta})$$

$$= \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_i (\boldsymbol{y} - \boldsymbol{\mu})^T V^{-1} (\boldsymbol{y} - \boldsymbol{\mu})$$

Set $\frac{\partial S_w(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -2X^T V^{-1}(\boldsymbol{y} - X\boldsymbol{\beta}) = 0$

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

If homoscedasticity

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

If homoscedasticity

- $\widehat{\boldsymbol{\beta}} = (X^T X)^{-1} X^T \boldsymbol{y}$
- $E\left(\widehat{\boldsymbol{\beta}}\right) = \boldsymbol{\beta}$
- $Var\left(\widehat{\boldsymbol{\beta}}\right) = \sigma^2 (X^T X)^{-1}$
- LSE is BLUE (Gauss-Markov)
- $\hat{\sigma}^2 = \frac{\hat{e}^T \hat{e}}{n-p}$

Coefficient of Determination

- $TSS = ESS + RSS$

$$\sum_{i}^{n}(y_i - \bar{y})^2 = \sum_{i}^{n}(y_i - \hat{y}_i)^2 + \sum_{i}^{n}(\hat{y}_i - \bar{y})^2$$

$$TSS = \boldsymbol{y}^T\boldsymbol{y} \ - n(\bar{y})^2$$

$$ESS = \boldsymbol{y}^T\boldsymbol{y} \ - \boldsymbol{y}^T\widehat{\boldsymbol{y}}$$

$$RSS = \widehat{\boldsymbol{y}}^T\widehat{\boldsymbol{y}} - n(\bar{y})^2$$

$$\boldsymbol{y}^T\widehat{\boldsymbol{y}} = \widehat{\boldsymbol{y}}^T\widehat{\boldsymbol{y}}$$

- $\hat{R}^2 = \frac{RSS}{TSS} = \left(r_{\boldsymbol{y},\widehat{\boldsymbol{y}}}\right)^2$

Categorical Features

Categorical Features

- J levels
- Use a coding scheme to encode it
  - Treatment (Dummy) coding
    - Dummy variables indicate the presence of levels
    - Effect beyond that of the reference level
  - Sum (Deviation) coding
    - Coefficients on coded variables sum to zero
    - Effect beyond the grand mean

Treatment Coding

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_A I_A + \hat{\beta}_B I_B + \hat{\beta}_C I_C + \hat{\beta}_x x$$

- Contrast matrix

strong multicollinearity
singular design matrix

$$\begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}$$

- If level A is present,

$$I_A = 1, I_B = I_C = 0$$
$$\hat{y}_A = \hat{\beta}_0 + \hat{\beta}_A + \hat{\beta}_x x$$

- $\hat{\beta}_A$ is the effect, the increase in $\hat{y}$, if level A is present, ceteris paribus.

Treatment Coding w/ a Reference Level

$$\hat{y} = (\hat{\beta}_0 + \hat{\beta}_B) + \hat{\beta}_A I_A + \hat{\beta}_C I_C + \hat{\beta}_x x$$

- Contrast matrix

$$\begin{matrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{matrix}$$

- If level A is present,

$$I_A = 1, I_C = 0$$
$$\hat{y}_A = (\hat{\beta}_0 + \hat{\beta}_B) + \hat{\beta}_A + \hat{\beta}_x x$$
$$\hat{\beta}_A = \hat{y}_A - \hat{y}_B$$

- $\hat{\beta}_A$ is the effect beyond that of level B, if level A is present, ceteris paribus.

Sum-to-Zero Coding (1)

$$\hat{y} = GM + \hat{\beta}_A I_A + \hat{\beta}_B I_B + \hat{\beta}_x x$$

- GM is the mean of the group/category/level means
- Contrast matrix

$$\begin{matrix} 1 & 0 \\ 0 & 1 \\ -1 & -1 \end{matrix}$$

  – The level C coding forces coefficients to sum to zero

Sum-to-Zero Coding (2)

$$\hat{y} = GM + \hat{\beta}_A I_A + \hat{\beta}_B I_B + \hat{\beta}_x x$$

- If level A is present,

$$I_A = 1, I_B = 0$$
$$\hat{y}_A = GM + \hat{\beta}_A + \hat{\beta}_x x$$

- $\hat{\beta}_A$ is the effect beyond GM, if level A is present, ceteris paribus.

Sum-to-Zero Coding (3)

$$\hat{y} = GM + \hat{\beta}_A I_A + \hat{\beta}_B I_B + \hat{\beta}_x x$$

- If level C is present,

$$I_A = -1, I_B = -1$$
$$\hat{y}_A = GM - \hat{\beta}_A - \hat{\beta}_B + \hat{\beta}_x x$$
$$"\hat{\beta}_C" = -(\hat{\beta}_A + \hat{\beta}_B)$$

- $-\left(\hat{\beta}_A + \hat{\beta}_B\right)$ is the effect beyond GM, if level C is present

Detecting & Modeling Interaction Effects

Between Quantitative & Categorical

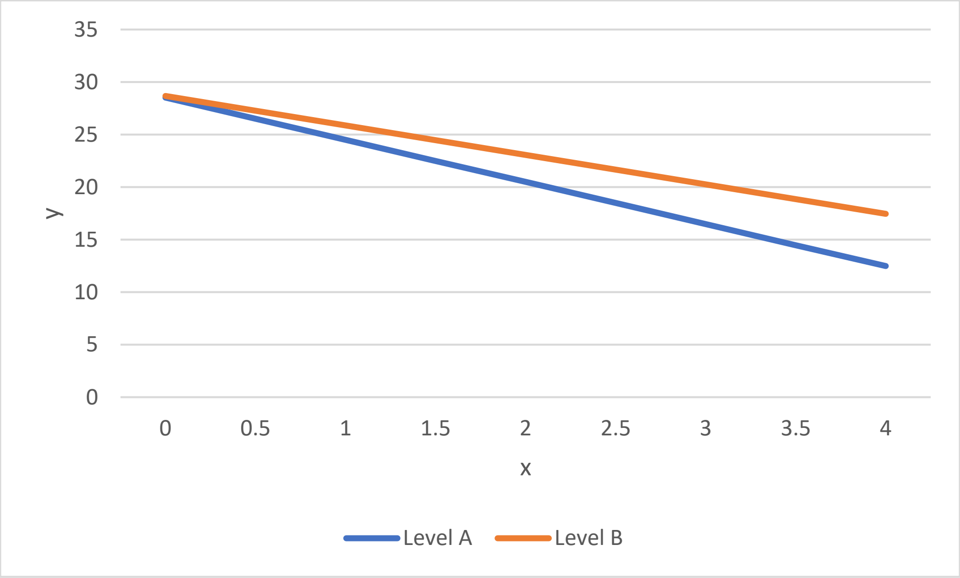
- Model

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_x x + \hat{\beta}_B I_B + \hat{\beta}_{xB} x I_B$$

- Margin is dependent on x

$$\hat{y}_B - \hat{y}_A = \hat{\beta}_B + \hat{\beta}_{xB} x$$

- example



Between Categorical &Categorical

- Model

$$y \sim \ldots + FactorA + FactorB + FactorA\!:\!FactorB$$

- example

```
SEX
FEMALE    44.188130
MALE      44.186982
Name: PRESTG10, dtype: float64

SEX     MARITAL
FEMALE  MARRIED          45.994792
        NEVER MARRIED    40.905363
MALE    MARRIED          47.692449
        NEVER MARRIED    37.884106
Name: PRESTG10, dtype: float64
```

Between Quantitative & Quantitative

- Model

$$y \sim \ldots + x_1 + x_2 + x_1 : x_2$$

- example

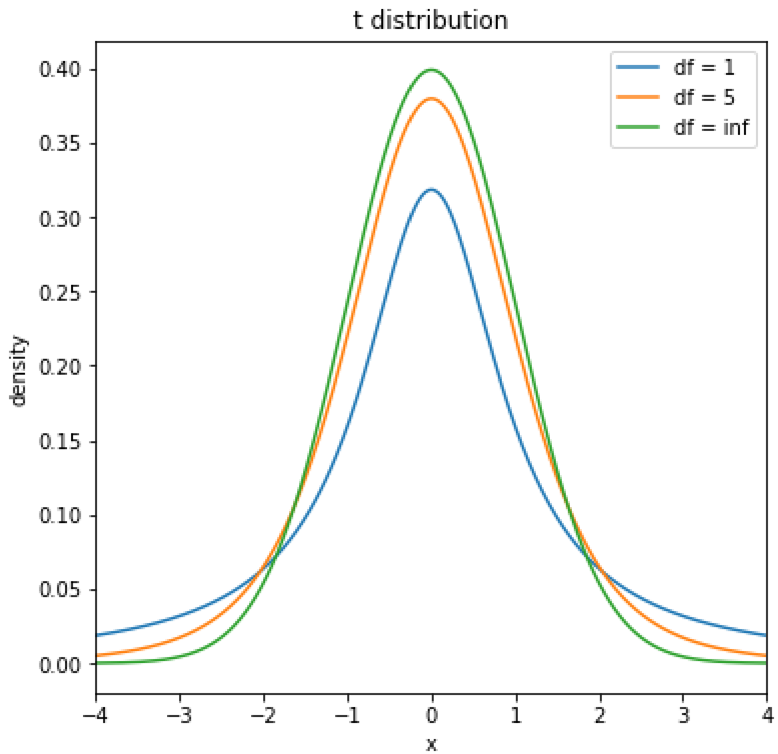
$$\begin{aligned}\hat{y} &= \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_{12} x_1 x_2 \\ &= \hat{\beta}_0 + \left(\hat{\beta}_1 + \hat{\beta}_{12} x_2\right) x_1 + \hat{\beta}_2 x_2 \\ &= \hat{\beta}_0 + \hat{\beta}_1 x_1 + (\hat{\beta}_2 + \hat{\beta}_{12} x_1) x_2\end{aligned}$$

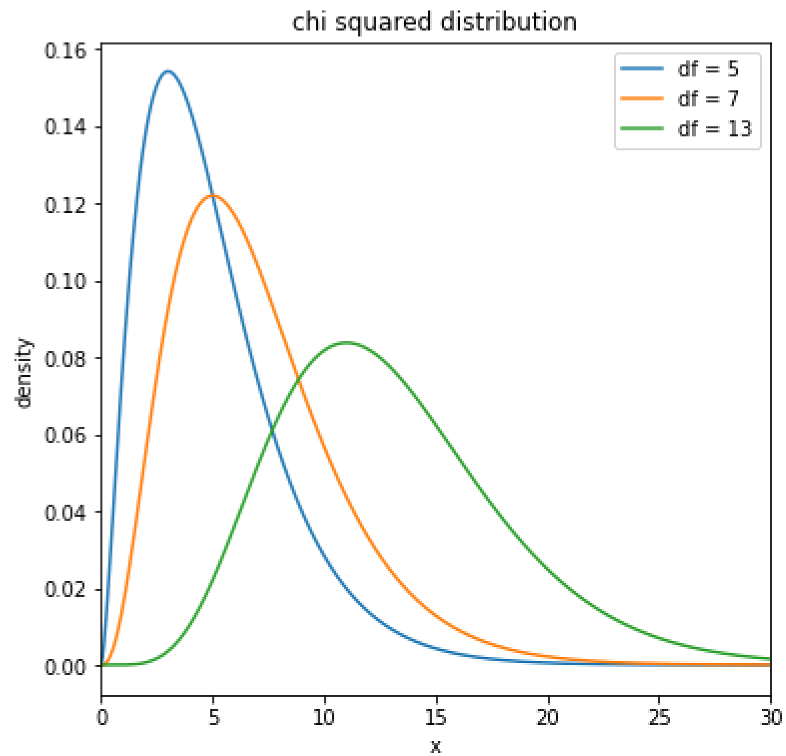
  - For $\hat{\beta}_{12} > 0$,
    - the greater the x_2, the more positive the effect of x_1 on the response is
    - the greater the x_1, the more positive the effect of x_2 on the response is

Normal Linear Model

- Assume
$$\boldsymbol{y} = (Y_1, \dots, Y_n)^T \sim \mathcal{N}(X\boldsymbol{\beta}, \sigma^2 I)$$
- $\widehat{\boldsymbol{\beta}} \sim \mathcal{N}(\boldsymbol{\beta}, (X^T X)^{-1}\sigma^2)$
- $\frac{\hat{\epsilon}^T \hat{\epsilon}}{\sigma^2} = \frac{(n-p)\,\hat{\sigma}^2}{\sigma^2} \sim \chi^2_{n-p}$
- $t = \frac{\widehat{\beta}_j - \beta_j}{\hat{\sigma}\sqrt{(X^T X)^{-1}_{jj}}} \sim t_{n-p}$

T and χ^2 Distributions





NLM: inference on a coefficient

- $H_0: \beta_j = d$ vs
- $H_1: \beta_j \neq d \rightarrow |t| > t_{n-p,1-0.5\alpha}$
- $H_1: \beta_j > d \rightarrow t > t_{n-p,1-\alpha}$
- $H_1: \beta_j < d \rightarrow t < -t_{n-p,1-\alpha}$
  - where $t = \frac{\hat{\beta}_j - d}{\hat{\sigma}\sqrt{(X^T X)^{-1}_{jj}}} \sim t_{n-p}$
- $100(1-\alpha)\%$ confidence interval for $\hat{\beta}_j$

$$\hat{\beta}_j \pm t_{n-p,1-\alpha/2}\, \hat{\sigma} \sqrt{(X^T X)^{-1}_{jj}}$$

NLM: inference for prediction

- $100(1-\alpha)\%$ confidence interval for the predicted mean function at $\boldsymbol{x}_0$

$$\boldsymbol{x}_0^T\widehat{\boldsymbol{\beta}} \pm t_{n-p,1-\alpha/2}\, \hat{\sigma}(\boldsymbol{x}_0^T(X^TX)^{-1}\boldsymbol{x}_0)^{1/2}$$

- $100(1-\alpha)\%$ prediction interval for the prediction at $\boldsymbol{x}$

$$\boldsymbol{x}_0^T\widehat{\boldsymbol{\beta}} \pm t_{n-p,1-\alpha/2}\, \hat{\sigma}(1+\boldsymbol{x}_0^T(X^TX)^{-1}\boldsymbol{x}_0)^{1/2}$$

Power Transforms

- To make the assumption of normality (if desired) more plausible

Strictly Positive Data

- Box-Cox family of transforms $(y > 0)$

$$y^{(\lambda)} = \begin{cases} \dfrac{y^{\lambda} - 1}{\lambda}, & \lambda \neq 0 \\ \ln y\,, & \lambda = 0 \end{cases}$$

- Estimate λ via maximum likelihood
- In practice,
  - Typically, $\lambda = 1, 0.5, 0, -1$
  - No need to -1 and $\div\,\lambda$ for operations unaffected by location and scale shifts (e.g. some regressions)

General Data

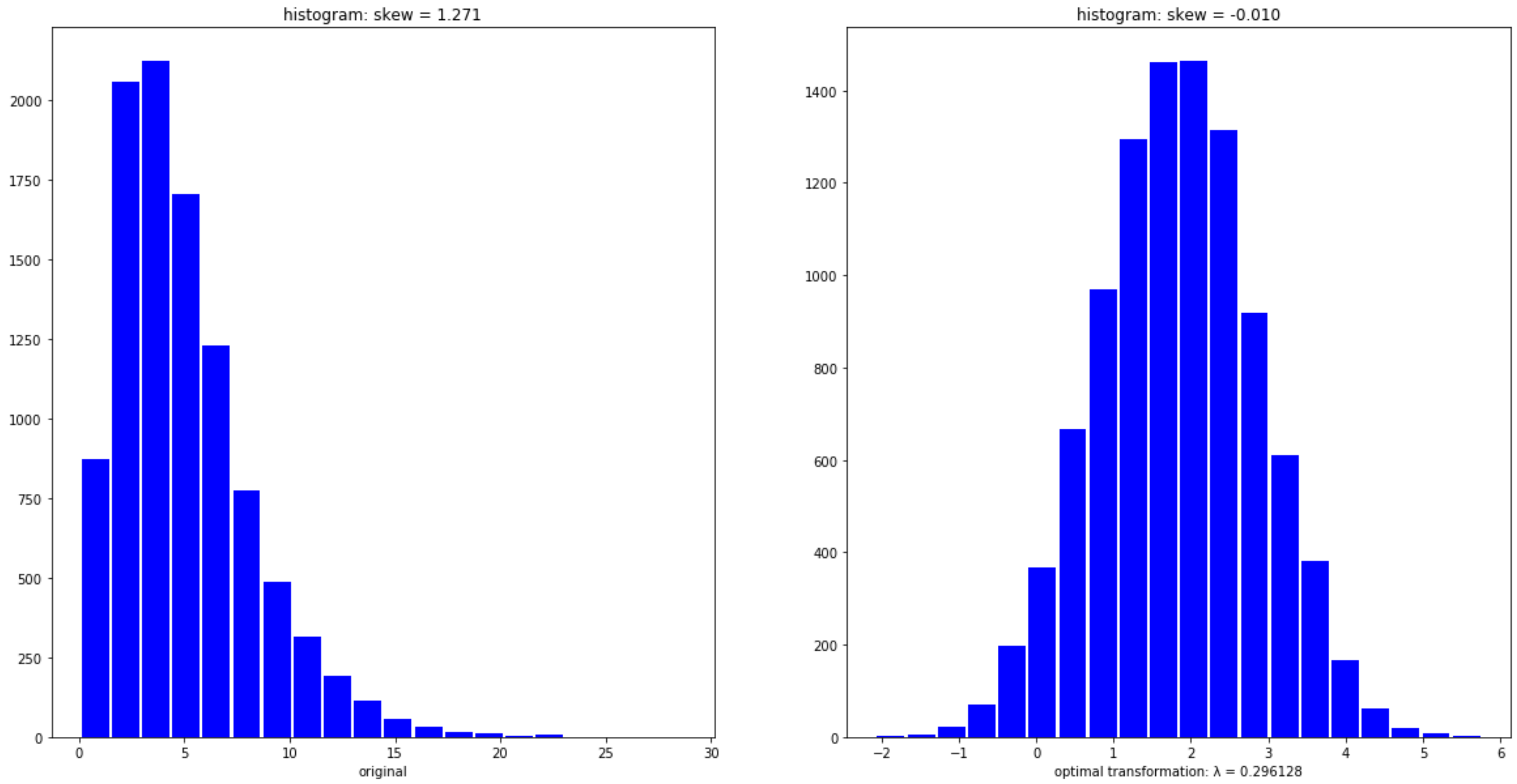
- Box-Cox family of transforms $(y > -\alpha)$

$$y^{(\lambda)} = \begin{cases} \dfrac{(y+\alpha)^{\lambda}-1}{\lambda}, & \lambda \neq 0 \\ \ln(y+\alpha), & \lambda = 0 \end{cases}$$

- Yeo-Johnson family of transforms

$$y^{(\lambda)} = \begin{cases} \dfrac{(y+1)^{\lambda}-1}{\lambda}, & \lambda \neq 0, y \geq 0 \\ \ln(y+1), & \lambda = 0, y \geq 0 \\ -\dfrac{(-y+1)^{2-\lambda}-1}{2-\lambda}, & \lambda \neq 2, y < 0 \\ -\ln(-y+1), & \lambda = 0, y < 0 \end{cases}$$

Box-Cox Transform Example

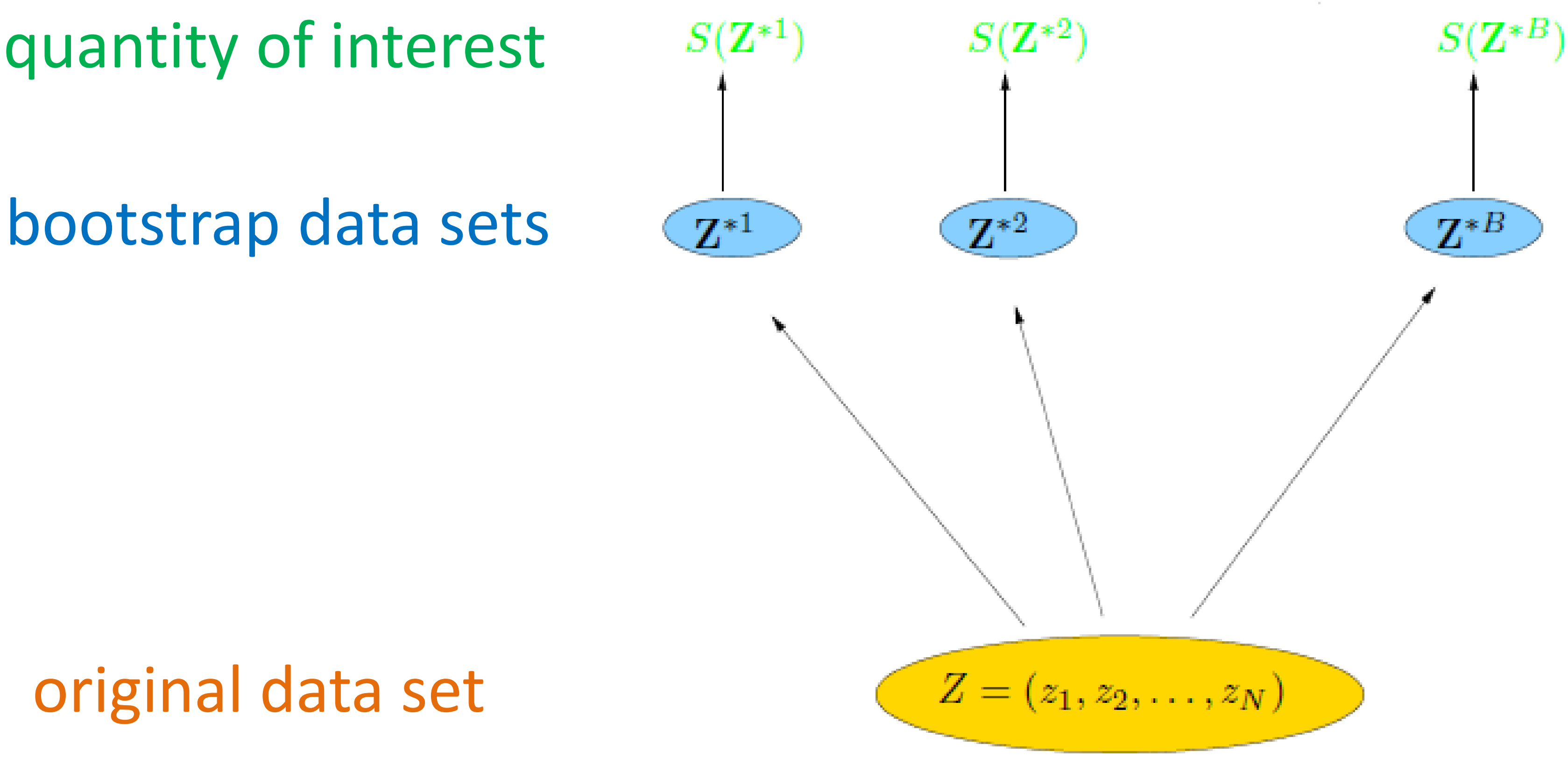


sklearn.preprocessing.PowerTransformer()

The Bootstrap

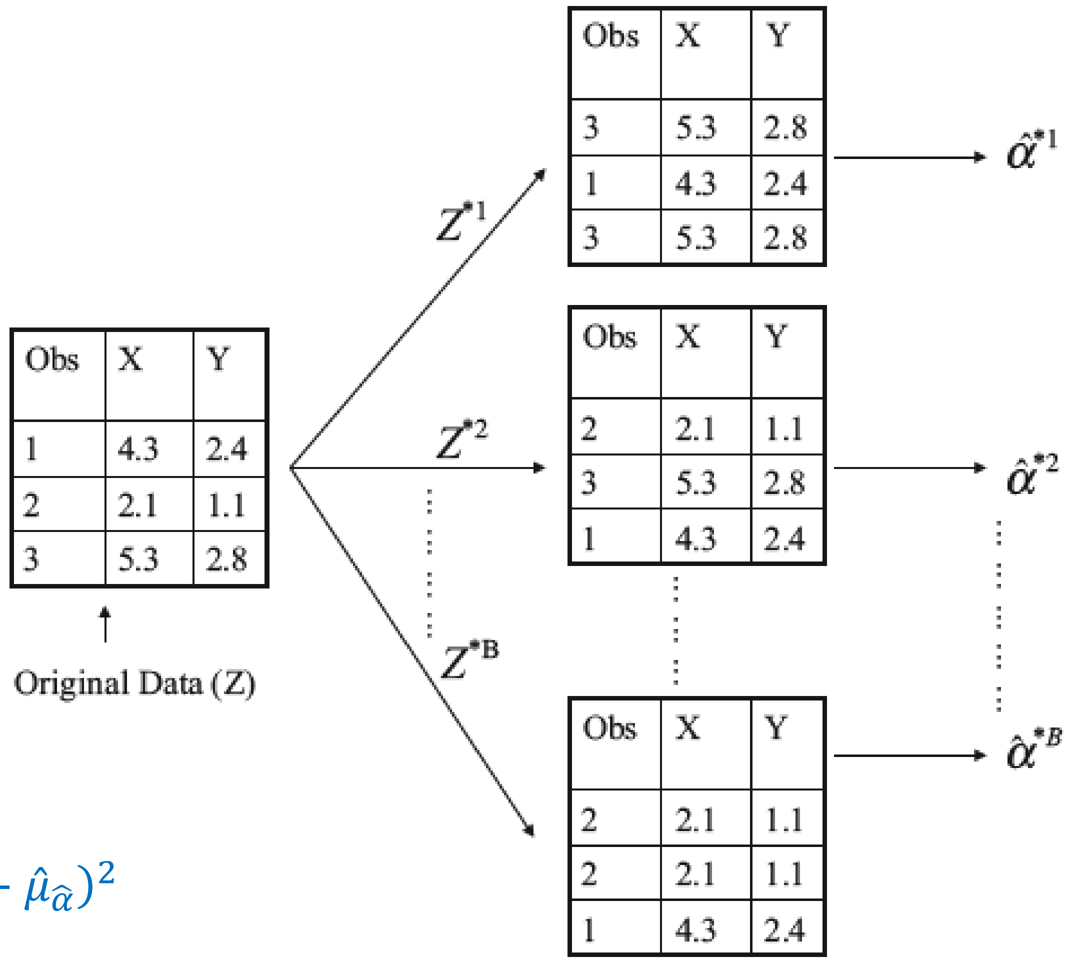
- Emulate the process of obtaining new samples without generating additional samples
- Obtain distinct data sets by repeatedly sampling, with replacement, observations from the original data set
- Works well if the original data set is a good representation of the population
  - Risk of “garbage in, garbage out”

The Bootstrap



Bootstrap Data Sets

$$\hat{\mu}_{\hat{\alpha}} = \frac{1}{B}\sum_{r=1}^{B} \hat{\alpha}^{*r}$$



Original Data (Z):

Obs	X	Y
1	4.3	2.4
2	2.1	1.1
3	5.3	2.8

Z^{*1}:

Obs	X	Y
3	5.3	2.8
1	4.3	2.4
3	5.3	2.8

Z^{*2}:

Obs	X	Y
2	2.1	1.1
3	5.3	2.8
1	4.3	2.4

Z^{*B}:

Obs	X	Y
2	2.1	1.1
2	2.1	1.1
1	4.3	2.4

$$\hat{\sigma}^2_{\hat{\alpha}} = \frac{1}{B-1}\sum_{r=1}^{B} (\hat{\alpha}^{*r} - \hat{\mu}_{\hat{\alpha}})^2$$

Gradient

- Directional Derivative

$$D_{\boldsymbol{u}} f(\boldsymbol{x}) \equiv \lim_{h \to 0} \frac{1}{h} [f(\boldsymbol{x} + h\boldsymbol{u}) - f(\boldsymbol{x})]$$

$$= \boldsymbol{\nabla} f(\boldsymbol{x}) \cdot \boldsymbol{u} = |\boldsymbol{\nabla} f(\boldsymbol{x})||\boldsymbol{u}| cos\theta = |\boldsymbol{\nabla} f(\boldsymbol{x})| cos\theta$$

where

$$\boldsymbol{\nabla} f = \left(\frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_p} \right)^T$$

$$\boldsymbol{u} = \left(u_1, \ldots, u_p \right)^T, \qquad \|\boldsymbol{u}\| = 1$$

- $D_{\boldsymbol{u}} f(\boldsymbol{x})$ is maximized at $|\boldsymbol{\nabla} f(\boldsymbol{x})|$ when $\boldsymbol{u}$ has the same direction as $\boldsymbol{\nabla} f(\boldsymbol{x})$.

$D_{\boldsymbol{u}} f(\boldsymbol{x}) = \nabla f(\boldsymbol{x}) \cdot \boldsymbol{u}$

$$g(h) = f(\boldsymbol{x} + h\boldsymbol{u}) = \begin{cases} f\left(x_1 + hu_1, \ldots, x_p + hu_p\right), & [A] \\ f\left(x_1^*, \ldots, x_p^*\right), & [B] \end{cases}$$

- From [A]

$$g'(0) = \lim_{s \to 0} \frac{1}{s}[g(0+s) - g(0)]$$
$$= \lim_{s \to 0} \frac{1}{s}[f(\boldsymbol{x} + (0+s)\boldsymbol{u}) - f(\boldsymbol{x} + 0\boldsymbol{u})] = D_{\boldsymbol{u}} f(\boldsymbol{x})$$

- From [B]

$$g'(h) = \frac{dg(h)}{dh} = \frac{\partial f(\boldsymbol{x}^*)}{\partial x_1^*}\frac{\partial x_1^*}{\partial h} + \cdots + \frac{\partial f(\boldsymbol{x}^*)}{\partial x_p^*}\frac{\partial x_p^*}{\partial h}$$
$$= \frac{\partial f(\boldsymbol{x}^*)}{\partial x_1^*} u_1 + \cdots + \frac{\partial f(\boldsymbol{x}^*)}{\partial x_p^*} u_p$$
$$\to g'(0) = \frac{\partial f(\boldsymbol{x})}{\partial x_1} u_1 + \cdots + \frac{\partial f(\boldsymbol{x})}{\partial x_p} u_p = \nabla f(\boldsymbol{x}) \cdot \boldsymbol{u}$$

Maximum Rate of Change

- When $\nabla f(\boldsymbol{x}) \neq \mathbf{0}$
  - the maximum rate of increase of f is $|\nabla f(\boldsymbol{x})|$ and is in the direction of $\nabla f(\boldsymbol{x})$
  - the maximum rate of decrease of f is $|\nabla f(\boldsymbol{x})|$ and is in the direction of $-\nabla f(\boldsymbol{x})$

Example

- $f(\boldsymbol{x}) = x_1 + x_2^2, \boldsymbol{\nabla} f(\boldsymbol{x}) = \begin{pmatrix} 1 \\ 2x_2 \end{pmatrix}$
- The unit vector in the direction of the gradient at $\boldsymbol{x}_0 = (1,1)^T$ is $\boldsymbol{u} = \left(\frac{1}{\sqrt{5}}, \frac{2}{\sqrt{5}}\right)^T$ since

$$\boldsymbol{\nabla} f(\boldsymbol{x}_0) = (1,2)^T, \qquad |\boldsymbol{\nabla} f(\boldsymbol{x}_0)| = \sqrt{5}$$

- Moving d units from $\boldsymbol{x}_0 = (1,1)^T$ in the direction of the gradient vector will land at $\boldsymbol{x}_1 = \boldsymbol{x}_0 + \boldsymbol{u}d = \left(1 + \frac{d}{\sqrt{5}}, 1 + \frac{2d}{\sqrt{5}}\right)^T$, with

rate of increase

$$f(\boldsymbol{x}_1) = f(\boldsymbol{x}_0) + \sqrt{5}d + \frac{4d^2}{5}$$

predicted increase

"error" on large steps

Gradient Descent

- To minimize the loss function

$$R(\widehat{\boldsymbol{\beta}}) = \sum_{i=1}^{n} R_i(\widehat{\boldsymbol{\beta}}) = \sum_{i=1}^{n} \left(y_i - \hat{y}_i(\widehat{\boldsymbol{\beta}})\right)^2$$

with learning rate $\eta > 0$, updating involves all observations

$$\widehat{\boldsymbol{\beta}}^{(r+1)} = \widehat{\boldsymbol{\beta}}^{(r)} - \eta \boldsymbol{\nabla} R\left(\widehat{\boldsymbol{\beta}}^{(r)}\right)$$

$$= \widehat{\boldsymbol{\beta}}^{(r)} - \eta \sum_{i=1}^{n} \boldsymbol{\nabla} R_i(\widehat{\boldsymbol{\beta}}^{(r)})$$

Stochastic Gradient Descent

- SGD is a stochastic approximation of GD.
- SGD uses randomly selected samples/subset from the training set for each iteration
- At extreme, updating would involve only a single (randomly selected) observation

$$\widehat{\boldsymbol{\beta}}^{(r+1)} = \widehat{\boldsymbol{\beta}}^{(r)} - \eta \boldsymbol{\nabla} R_i\left(\widehat{\boldsymbol{\beta}}^{(r)}\right)$$

Gradient Descent for Linear Regression

- Loss function

$$R(\widehat{\boldsymbol{\beta}}) = \sum_{i=1}^{n} R_i(\widehat{\boldsymbol{\beta}}) = \sum_{i=1}^{n} \left(y_i - \boldsymbol{x}_i^T \widehat{\boldsymbol{\beta}}\right)^2$$

- Gradient

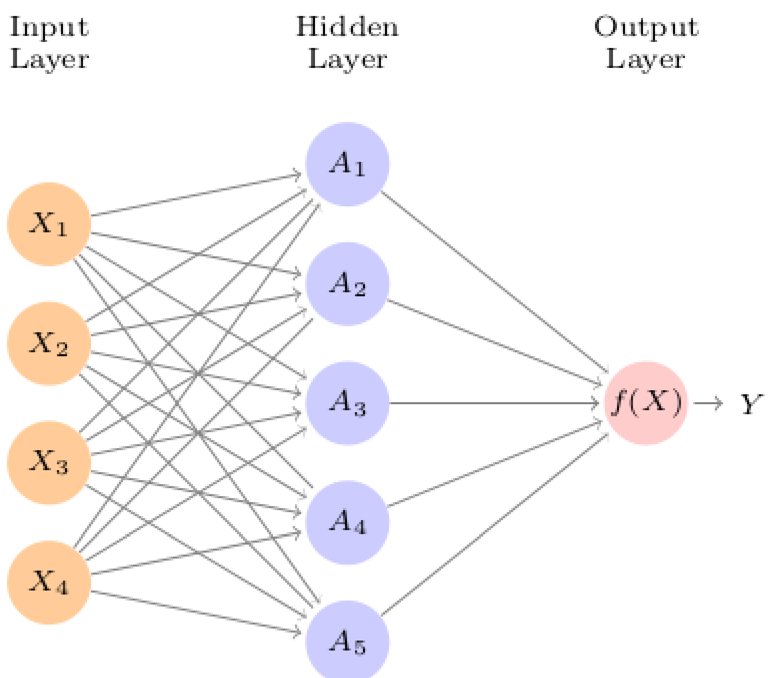
$$\boldsymbol{\nabla} R_i(\widehat{\boldsymbol{\beta}}) = -2\left(y_i - \boldsymbol{x}_i^T \widehat{\boldsymbol{\beta}}\right)\boldsymbol{x}_i = -2\hat{e}_i\left(\widehat{\boldsymbol{\beta}}^{(r)}\right)\boldsymbol{x}_i$$

- Updating

$$\widehat{\boldsymbol{\beta}}^{(r+1)} = \widehat{\boldsymbol{\beta}}^{(r)} - \eta \boldsymbol{\nabla} R\left(\widehat{\boldsymbol{\beta}}^{(r)}\right)$$

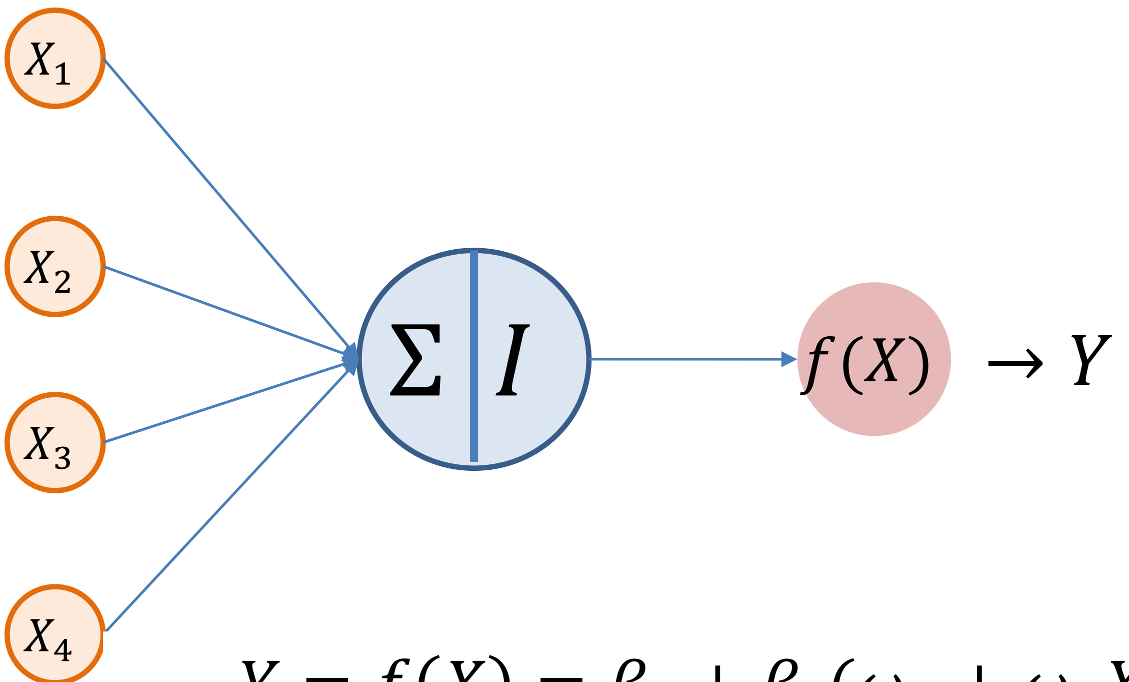
$$= \widehat{\boldsymbol{\beta}}^{(r)} + 2\eta \sum_{i=1}^{n} \hat{e}_i\left(\widehat{\boldsymbol{\beta}}^{(r)}\right)\boldsymbol{x}_i$$

Single Layer Neural Network



$$\hat{y} = f(X) = \beta_0 + \sum_{k=1}^{K} \beta_k A_k \, , \qquad A_k = g(\boldsymbol{x}_i^T \boldsymbol{\omega}_k)$$

NN: Linear Regression



$$
\begin{aligned}
Y = f(X) &= \beta_0 + \beta_1(\omega_0 + \omega_1 X_1 + \cdots + \omega_p X_p) \\
&= (\beta_0 + \beta_1\omega_0) + \beta_1\omega_1 X_1 + \cdots + \beta_p\omega_p X_p \\
&= b_0 + b_1 X_1 + \cdots + b_p X_p
\end{aligned}
$$

Bayesian Approach

$$f_{\mathrm{D}|\Theta}(\boldsymbol{d}) = \frac{f_{\Theta,D}(\boldsymbol{\theta}, \boldsymbol{d})}{f_{\Theta}(\boldsymbol{\theta})}$$

$$f_{\Theta|D}(\boldsymbol{\theta}) = \frac{f_{\Theta,D}(\boldsymbol{\theta}, \boldsymbol{d})}{f_{\mathrm{D}}(\boldsymbol{d})} = \frac{f_{\mathrm{D}|\Theta}(\boldsymbol{d}) f_{\Theta}(\boldsymbol{\theta})}{f_{\mathrm{D}}(\boldsymbol{d})}$$

$$f_{\mathrm{D}}(\boldsymbol{d}) = \int f_{\mathrm{D}|\Theta}(\boldsymbol{d})\, d\boldsymbol{\Theta}$$

- Prior
- Likelihood
- Posterior

Example

- $\Theta = \{\mu, \sigma\}$
  - Assumption: μ and σ are independent

$$Y_i \sim^{iid} \mathcal{N}(\mu, \sigma)$$
$$\mu \sim \mathcal{N}(\mu_\mu, \sigma_\mu)$$
$$\sigma \sim \mathcal{U}(\sigma_L, \sigma_H)$$

Example: Likelihood

- Prior

$$f_{\Theta}(\boldsymbol{\theta}) = \prod_{j=1}^{p} f_{\Theta_j}(\theta_j) = \frac{\exp[-\left(\frac{\mu-\mu_\mu}{\sigma_\mu}\right)^2]}{\sqrt{2\pi}\sigma_\mu} \cdot \frac{1}{(\sigma_H - \sigma_L)}$$

- Likelihood

$$f_{\mathrm{D}|\Theta}(\boldsymbol{d}) = \prod_{i=1}^{n} \frac{\exp[-\left(\frac{y_i-\mu}{\sigma}\right)^2]}{\sqrt{2\pi}\sigma}$$

- Posterior ∝ Likelihood × Prior

$$f_{\Theta|D}(\boldsymbol{\theta}) = \frac{f_{\mathrm{D}|\Theta}(\boldsymbol{d}) f_{\Theta}(\boldsymbol{\theta})}{f_{\mathrm{D}}(\boldsymbol{d})} \propto f_{\mathrm{D}|\Theta}(\boldsymbol{d}) f_{\Theta}(\boldsymbol{\theta})$$

Example: Log Likelihood

- Posterior

$$f_{\Theta|D}(\boldsymbol{\theta}) = \frac{f_{\mathrm{D}|\Theta}(\boldsymbol{d}) f_{\Theta}(\boldsymbol{\theta})}{f_{\mathrm{D}}(\boldsymbol{d})}$$

- Log Posterior

$$\ln f_{\mathrm{D}|\Theta}(\boldsymbol{d}) + \ln f_{\Theta}(\boldsymbol{\theta}) + \text{const}$$

$$l(\boldsymbol{\theta} \mid \boldsymbol{d}) = -\sum_{i=1}^{n} \left(\frac{y_i - \mu}{\sigma}\right)^2 - \left(\frac{\mu - \mu_\mu}{\sigma_\mu}\right)^2 + \text{const}$$

- $\widehat{\boldsymbol{\theta}}$ maximizes $l(\boldsymbol{\theta} \mid \boldsymbol{d})$

Bayesian Regression

$$Y_i \sim^{iid} \mathcal{N}(\mu_i, \sigma)$$
$$\mu_i = \boldsymbol{x}_i^T \boldsymbol{\beta}$$
$$\beta_j \sim$$
$$\sigma \sim$$

Fitting a Bayesian Regression

- Numerically minimize $-l(\boldsymbol{\beta} \mid \text{data})$
  - GD-like?
- Numerically represent the posterior
  - Grid approximation
- Posterior Sampling
  - Markov Chain Monte Carlo

Linear Regression: Gibbs Sampling

- example

$$\beta_j \sim N\left(\mu_{\beta_j}, \sigma^2_{\beta_j}\right)$$
$$\sigma^2 \sim IG(a, b)$$

- Conditional posterior

$$\boldsymbol{\beta}|y, \sigma^2 \sim N((X'X)^{-1}X'\boldsymbol{y}, \sigma^2(X'X)^{-1})$$
$$\sigma^2|y, \boldsymbol{\beta} \sim IG(a + 0.5n, b + 0.5\boldsymbol{e}'\boldsymbol{e})$$

Linear Regression: Gibbs Sampling

- Gibbs Sampling

Initialize $\widehat{\boldsymbol{\beta}}^{(1)}$

Draw $\hat{\sigma}^{2^{(1)}}$ from $\sigma^2|y, \widehat{\boldsymbol{\beta}}^{(1)}$

Draw $\hat{\beta}^{(2)}$ from $\boldsymbol{\beta}|y, \hat{\sigma}^{2^{(1)}}$

Draw $\hat{\sigma}^{2^{(2)}}$ from $\sigma^2|y, \widehat{\boldsymbol{\beta}}^{(2)}$

Draw $\hat{\beta}^{(3)}$ from $\boldsymbol{\beta}|y, \hat{\sigma}^{2^{(2)}}$

…