

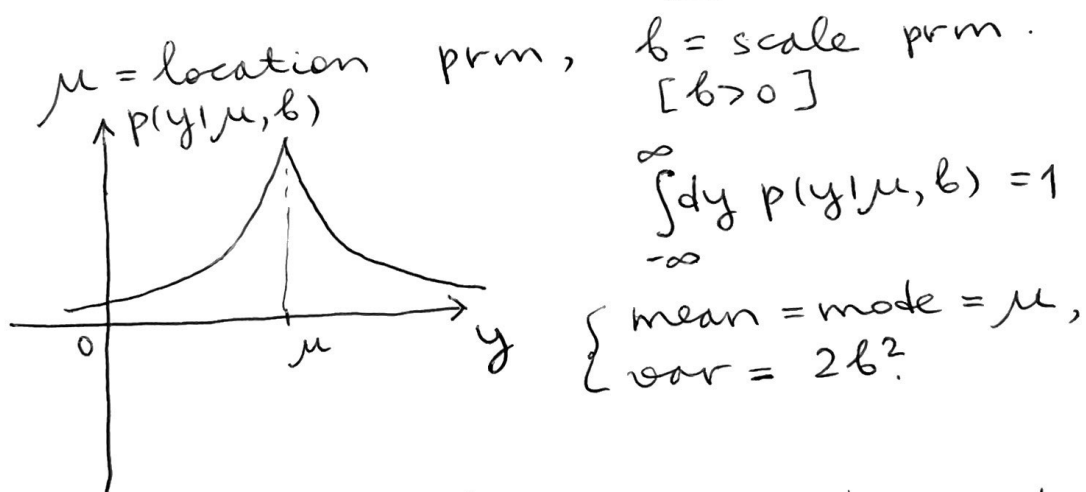
Lecture 13

Robust linear regression

Fitting gaussian models is sensitive to outliers $\Rightarrow$ one can use heavy-tail distributions to achieve robustness to outliers.

① Laplace likelihood

Laplace distribution: $p(y|\mu, b) = \frac{1}{2b} e^{-\frac{|y-\mu|}{b}}$



Needs techniques like projected grad for robust optimization.

② Student's t-likelihood

Student's t-distribution:

$$p(y|\mu, \sigma, \nu) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\pi\nu} \Gamma(\frac{\nu}{2})} \left(1 + \frac{1}{\nu} \underbrace{\left(\frac{y-\mu}{\sigma}\right)^2}_{t}\right)^{-\frac{\nu+1}{2}}$$

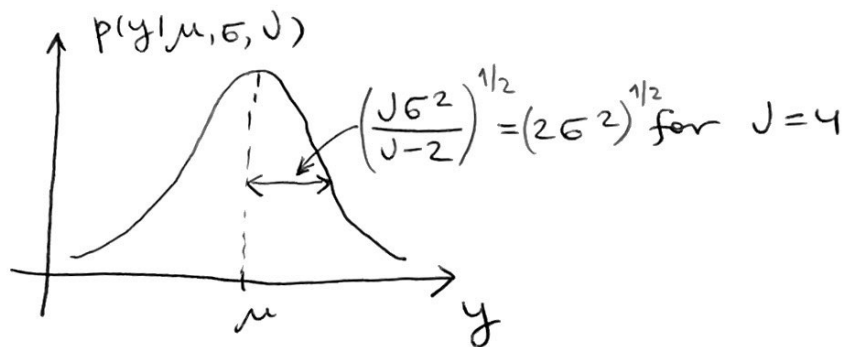
μ = mean, $\sigma > 0$ = scale prm,
prm $\nu > 0$ = DoF

$$\begin{cases} \text{mean} = \mu, & J > 1 \\ \text{mode} = \mu \\ \text{var} = \frac{J\sigma^2}{J-2}, & J > 2 \end{cases} \Rightarrow J=4 \text{ is commonly used}$$

Note that for $t^2 \gg J$,

$$p(t) \sim (t^2)^{-\frac{J+1}{2}} = t^{-(J+1)}$$

Thus, the tail decays much slower than the Gaussian tail.

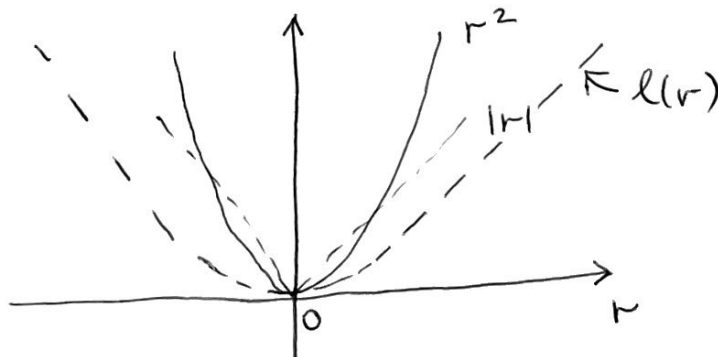


The model can be optimized using SGD or other standard techniques.

3. Huber loss

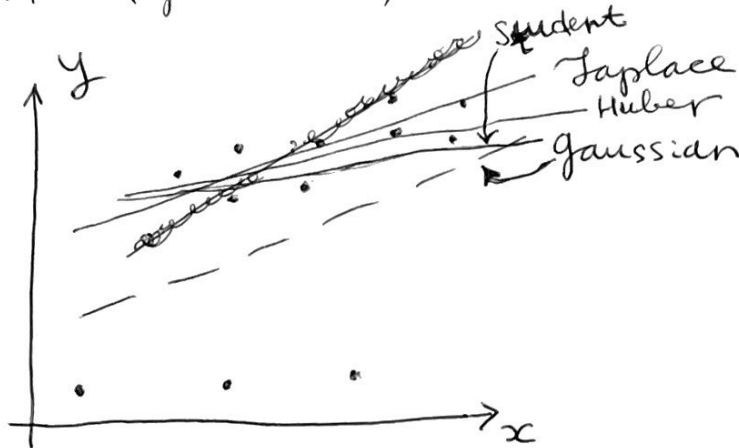
$$l(r) = \begin{cases} r^2/2, & |r| \leq \delta \\ \delta|r| - \delta^2/2, & |r| > \delta \end{cases}$$

fitting weight
 $r = w - a$
 $\uparrow$
 constraint, typically = 0,
 (or)
 $r = \text{obs.} - \text{model}$
 $y - \hat{y}$



Typically, $\delta = \mathcal{O}(1)$, e.g. 1.5

Huber loss can be used instead of l_2 (gaussian) loss or l_1 (Laplace) loss.



Bayesian linear regression

Likelihood: $p(\mathcal{D} | \vec{w}, \sigma^2) = \prod_{n=1}^N p(y_n | \vec{w} \cdot \vec{x}, \sigma^2) =$
 $= \mathcal{N}(\vec{y} | X \vec{w}, \sigma^2 \mathbb{I}_N)$. centered data

Prior: $p(\vec{w}) = \mathcal{N}(\vec{w} | \vec{0}, \tau^2 \mathbb{I}_D)$.

Then the posterior is given by

$p(\vec{w} | \mathcal{D}, \sigma^2) = \mathcal{N}(\vec{w} | \vec{w}', \Sigma')$, where

$$\begin{cases} \vec{w}' = \frac{1}{\sigma^2} \Sigma' X^T \vec{y}, \\ \Sigma' = \left(\frac{1}{\tau^2} \mathbb{I}_D + \frac{1}{\sigma^2} X^T X \right)^{-1} = \sigma^2 \left(\underbrace{\lambda \mathbb{I}_D}_{\frac{\sigma^2}{\tau^2}} + X^T X \right)^{-1}. \end{cases}$$

Then $\vec{w}' = (\lambda \mathbb{I}_D + X^T X)^{-1} X^T \vec{y}$,
 same as ridge regression.

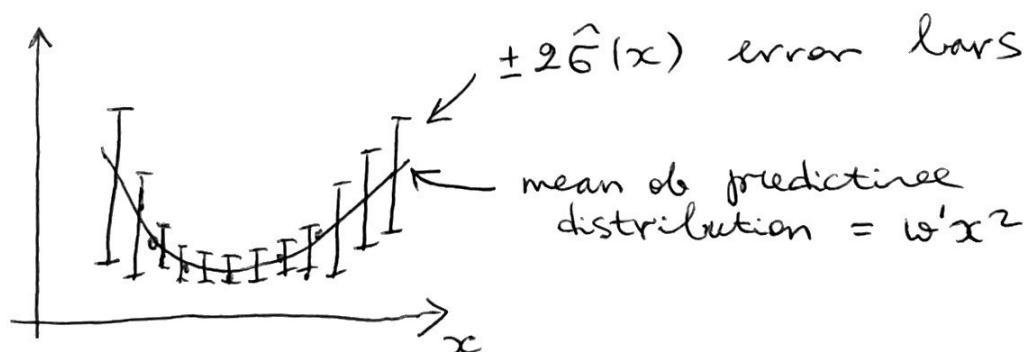
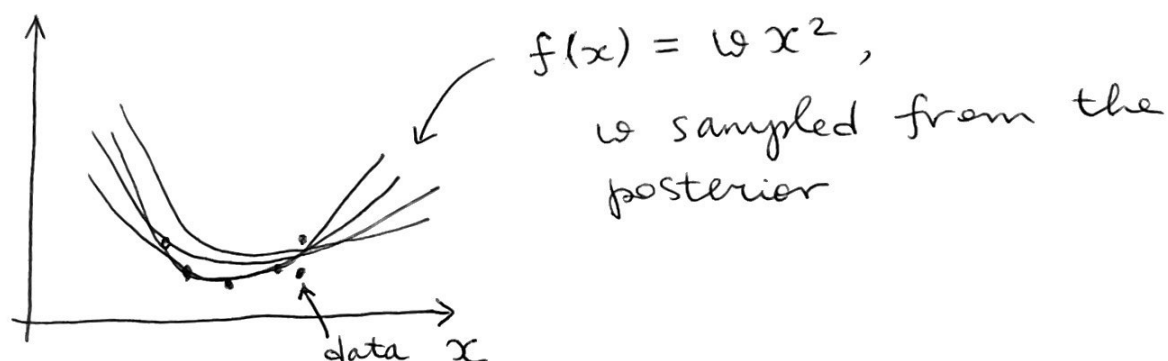
Finally, we can compute the predictive distribution:

$$p(y|\vec{x}, \mathcal{D}, \sigma^2) = \int d\vec{w} \mathcal{N}(\vec{y}|\vec{w} \cdot \vec{x}, \sigma^2) \times$$

$$\times \mathcal{N}(\vec{w}|\vec{w}', \Sigma') =$$

$$= \mathcal{N}(y|\vec{w}' \cdot \vec{x}, \hat{\sigma}^2(\vec{x})), \text{ where}$$

$$\hat{\sigma}^2(\vec{x}) = \sigma^2 + \vec{x}^T \Sigma' \vec{x}.$$



Now, we switch to features:

$$\begin{array}{ccc} \tilde{X}_{N \times D} & \Rightarrow & \Phi_{N \times M} \\ \text{data} & & \text{design} \\ \text{matrix} & & \text{matrix} \end{array}$$

and switch to Bishop's notation:

posterior $\Rightarrow p(\tilde{w} | \mathcal{D}, \alpha, \beta)$ becomes

$$p(\tilde{w} | \mathcal{D}, \underset{\tau^{-2}}{\alpha}, \underset{\sigma^{-2}}{\beta}) = \mathcal{N}(\tilde{w} | \bar{m}_N, S_N), \text{ where}$$

$$\mathcal{D} = \left\{ \underset{\substack{\uparrow \\ \text{instead of } \tilde{y}}}{\tilde{t}}, X \right\} \quad \begin{cases} \bar{m}_N = \beta S_N \Phi^T \tilde{t}, \\ S_N = (\alpha \mathbb{I} + \beta \Phi^T \Phi)^{-1} \end{cases}$$

Thus, we have relabeled:

$$\begin{cases} \tilde{w}' \Rightarrow \bar{m}_N, \\ \Sigma' \Rightarrow S_N. \end{cases}$$

Finally, the predictive distribution is given by

$$p(t | \tilde{x}, \mathcal{D}, \alpha, \beta) = \mathcal{N}(t | \bar{m}_N \cdot \underbrace{\tilde{\Psi}(\tilde{x})}_{\substack{\text{M-dim} \\ \text{feature vector}}}, \sigma_N^2(\tilde{x})),$$

where

$$\sigma_N^2(\tilde{x}) = \beta^{-1} + \tilde{\Psi}^T(\tilde{x}) S_N \tilde{\Psi}(\tilde{x}).$$

Note that as $N \rightarrow \infty$, we expect the posterior prob. to have vanishing variance: $S_N \rightarrow 0$ as $N \rightarrow \infty$.

Indeed, $(S_N^{-1})_{ij} \propto \beta (\Phi^T \Phi)_{ij} =$

$$= \beta \sum_{n=1}^N \varphi_i(\vec{x}_n) \varphi_j(\vec{x}_n).$$

This $\uparrow$ as $N \uparrow \Rightarrow S_N \downarrow$.

Moreover, if $\varphi(\vec{x})$ is localized,

$\sigma_N^2(\vec{x}) \rightarrow \beta^{-1}$ even for small N ,
if $\vec{x}$ is far away from the basis
function centers. Thus, the model
may become overconfident when
extrapolating.