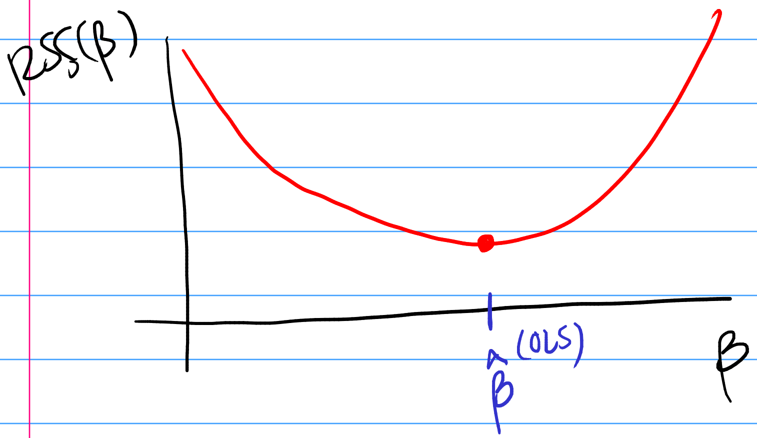
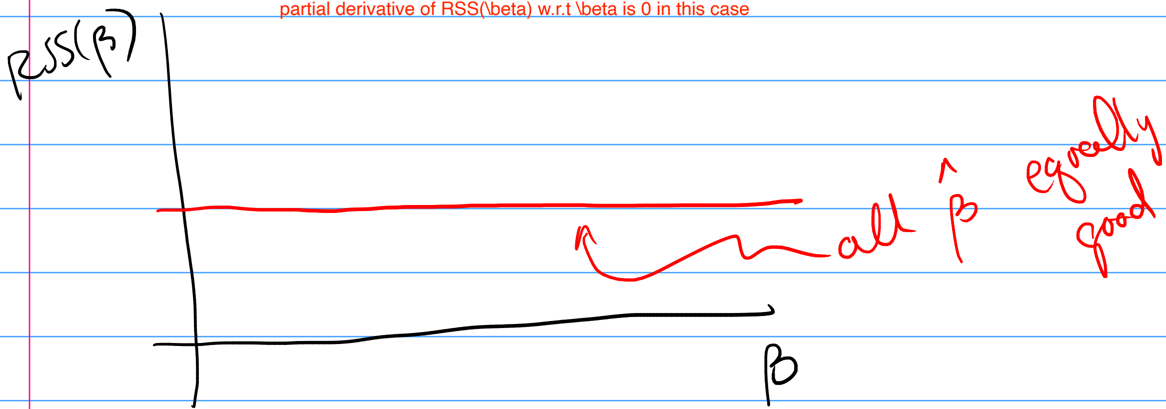
Lecture 4:

(1) X^TX is invertible

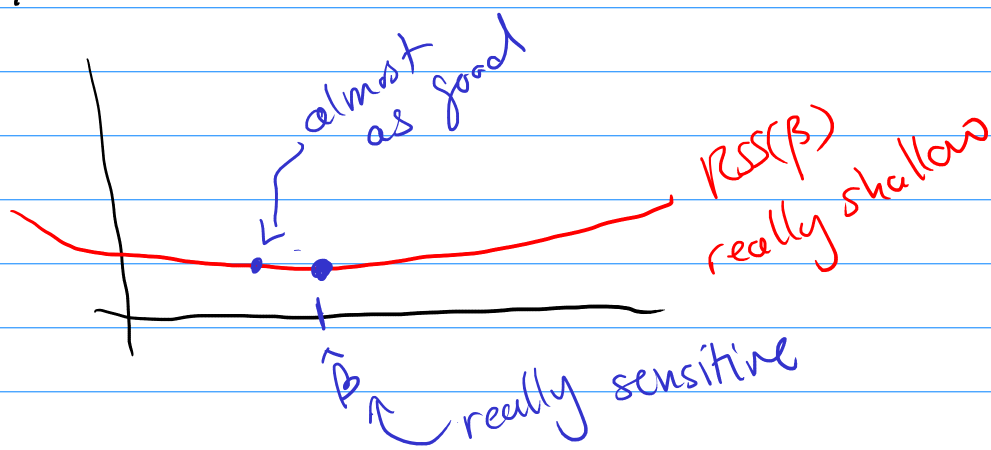


(2) X^TX not invertible

Check that \partial RSS(\beta)/ \partial \beta =0 , i.e., the partial derivative of RSS(\beta) w.r.t \beta is 0 in this case



(3) X^TX close to not invertible

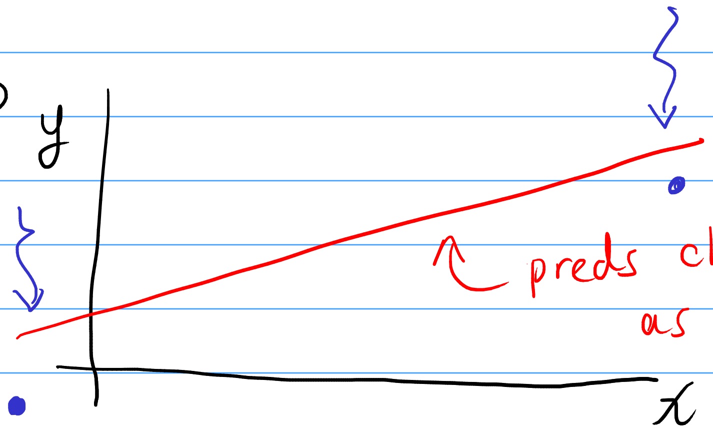


KNN Regression (k-nearest neighbors regression)

Recall for LR: we have a really strong global assumption about f

$$f(\underset{\sim}{x}) = \underline{x}^T \beta$$

e.g. in 1D



$\hat{f}(x) = \hat{\beta}x$

preds change by $\hat{\beta}$ as I move along x

Furthermore: training data affects the fit far away

Benefit: strong global assumption make $\hat{f}$ practical to find

---

kNN makes a weaker local assumption that values of $\hat{f}(\underset{\sim}{x})$ only depend on nearby training points.

k = integer that determines how many "nearby" training points we consider

Given training data $\{(\underset{\sim}{x}_n, y_n)\}_{n=1}^{N}$

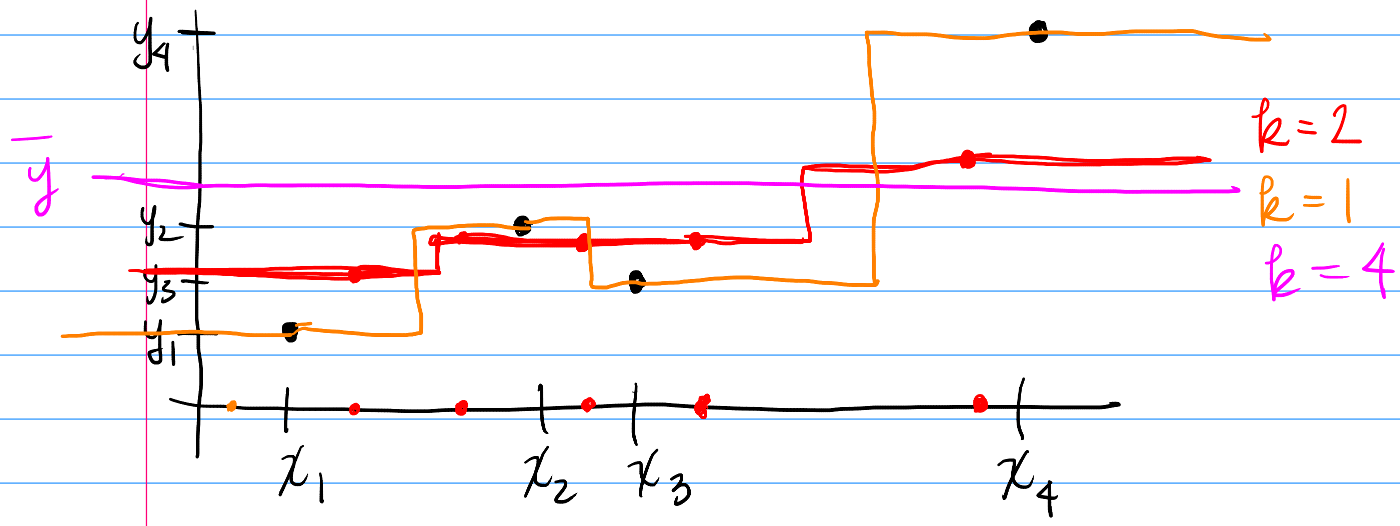
kNN fits $\hat{f}$ as

$$\hat{f}(\underset{\sim}{x}) = \frac{1}{k} \sum_{n \in N_k(\underset{\sim}{x})} y_n$$

= avg. y_n over k nearest neighbors to $\underset{\sim}{x}$

$N_k(\underset{\sim}{x})$ = k neighborhood of $\underset{\sim}{x}$
= indices of k nearest training pts to $\underset{\sim}{x}$

for numeric typically use euclidean



What happens as we change k?

general rule: k controls the flexibility of the method

↳ how complicated of a function $\hat{f}$ is

Small k → very flexible method
e.g. $k=1$ interpolates data

large k → very inflexible method
e.g. $k=N$ ⇒ just predict $\bar{y}$

Comparison w/ OLS:

OLS reduces $\mathcal{F}$ to a p-dim'l space.

kNN reduces $\mathcal{F}$ to a N/k dim'l space.

as $k \uparrow$ I reduce the dim of the space

(VC dimension Vapnik-Chervonensk)

converse: $k \downarrow$ I increase the dim