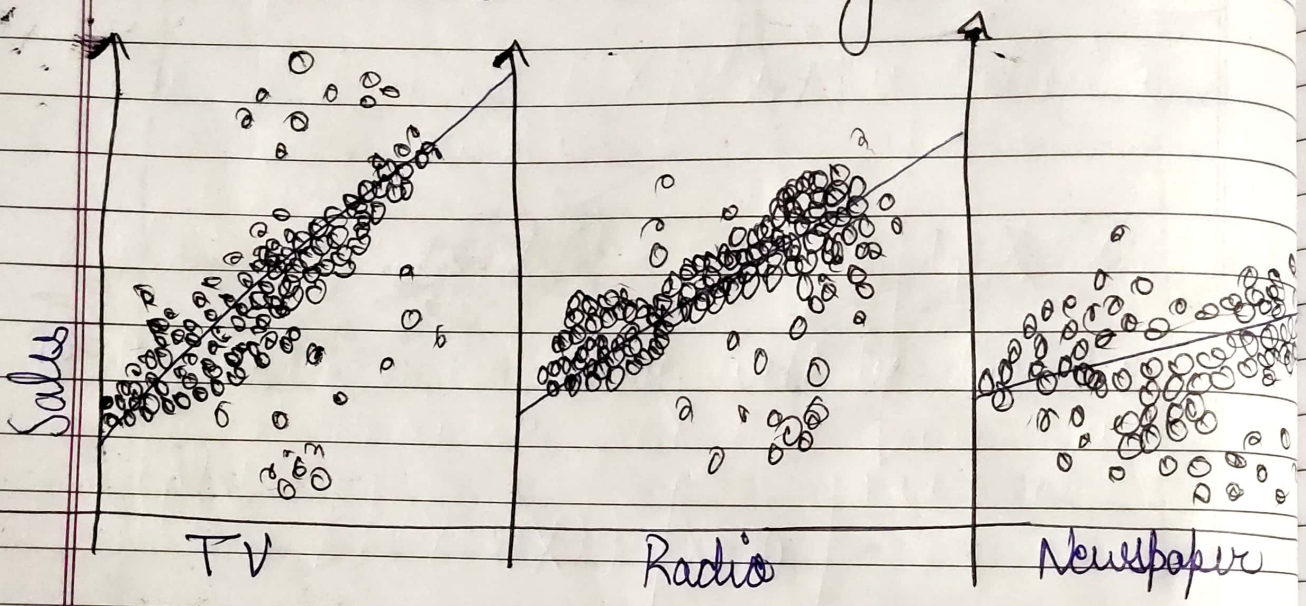
Date :
Page :

Ch=>2

What is Statistical Learning



Shown above is the Sales V/s TV, Radio & Newspaper with a blue linear regression line fit seprately to each

But can we predict Sales using all of these three.

The answer is "Yes". The Statistical learning helps you do so.

We here like to know how these feats together with each other i.e.

$$Sales \approx f(TV, Radio, Newspaper)$$

Here, Sales is a response / Target or Y.

And Input Variable/Features "X".

where, $\begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}$ represents TV
X_2 represents Radio
X_3 represents Newspaper

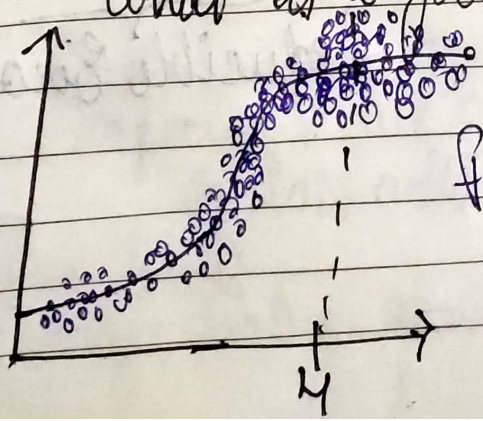
So input Vector $X = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}$

Now, we can write $\boxed{Y = f(x) + \varepsilon}$

where ε is an irreducible error / measurement error

→ So, with a good "f"
→ we can do good predictions
→ we can understand which input features are important.
→ we can also understand how each component X_j of X impacts Y.

→ Is there an ideal "f(x)"
→ If "yes" what is a good value of Y at X=4.



So, a good value for f(4) could be
⇒ E(Y|X=4)

$f(4) = E(Y|X=4)$ what does this specify?

This specifies $E(Y|X=4)$ means expected value / weighted average of value Y at $X=4$.

* This ideal $f(x) = E(Y|X=x)$ is called a regression function.

→ $f(x) = f(x_1, x_2, x_3) = E(Y|X_1=x_1, X_2=x_2, X_3=x_3)$ here,

→ The $f(x)$ presented above will be the ideal/best optimal predictor for Y that minimizes MSE

$$MSE = E[(Y-f(x))^2 | X=x] \quad ***$$

→ $\varepsilon = Y - f(x)$ is the irreducible error - i.e. even if we still make errors in predictions. Because even at $X=x$ there is a cloud/Distribution of Y values.

→ We represent our estimate by $\hat{f}(x)$ where ideal estimate is $f(x)$

$$E[(Y-\hat{f}(x))^2 | X=x] = \underbrace{[f(x)-\hat{f}(x)]^2}_{\text{Reducible Error}} + \underbrace{Var(\varepsilon)}_{\text{Irreducible Error}}$$

*** This Topic is Irrelevant for Non Stats Background ***

Proof: $E[(Y-\hat{f}(x))^2 | X=x] = (f(x)-\hat{f}(x))^2 + Var(\epsilon)$

add & subtract $f(x)$ to the LHS.

$$E[(Y+f(x)-f(x)-\hat{f}(x))^2 | X=x]$$

$$= E[(Y-E[Y|x])^2] + [E[(Y|x) - \hat{f}(x)]]^2$$
$$+2E[(Y-E(Y|x))(E[Y|x]-\hat{f}(x))]$$

We will let the first 2 terms stay, & deal with 3rd one.

In 3rd term we can apply tower property of conditional expectation that states

$$E(Y) = E[E(Y|X=x)]$$

so, $E[(Y-E[Y|X])(E[Y|x]-\hat{f}(x))]$

$$= E[(E[(Y-E[Y|x])|X])(E[Y|x]-\hat{f}(x))]$$

This will be reduced to zero as, Expected Value of Expected Value of loss will be zero.

Or in laymans terms average value of average loss will be zero

so the final term,

$$E\left[(Y - E[Y|X])^2\right] + E\left[(E[Y|X] - \hat{f}(x))^2\right]$$

Irreducible error — Reducible error

This will the Variance across of Y at a given x at all X's.

i.e. $E(\varepsilon) = Var(\varepsilon)$

because mean is 0

& $Var(\varepsilon) = E(\varepsilon^2) + (E(\varepsilon))^2$

this is 0

$$E\left((f(x) - \hat{f}(x))^2\right)$$

$$= (f(x) - \hat{f}(x))^2$$

because this is linear

* hint take 2 linear equations & test

So, $E\left[(Y - \hat{f}(x))^2 \mid X = x\right]$

$$= Var\,\varepsilon + (f(x) - \hat{f}(x))^2$$

xxx

xxx

Date :
Page :

Typically we've a few values of Y at $X=4$
so, we can't find exact predictor of Y but we can predict Y with good accuracy

→ Typically we've a few values at $X=4$
→ So we can't compute $E(Y|X=x)$

So, what can we do.

$$\hat{f}(x) = Ave(Y|x \in N(x))$$
where $N(x)$ is neighborhood of x.

Nearest Neighbor can be pretty good when
→ $p \leq 4$ (less predictors)
&
→ $n \gg p$ (large n), so that we've a high density neighborhood.

→ it might not work good when p is large.

Reason: Curse of Dimensionality: NearestNeighbor tend to be far in high dimensions.

→ We need to get a reasonable fraction of N values of y_i to bring the variance down. ex: 10%.
→ A 10% neighborhood in high dimensions will no longer be local, hence contradicting $E(Y|X=x)$.

So, what should we rather do?

Parametric & Structured Models:

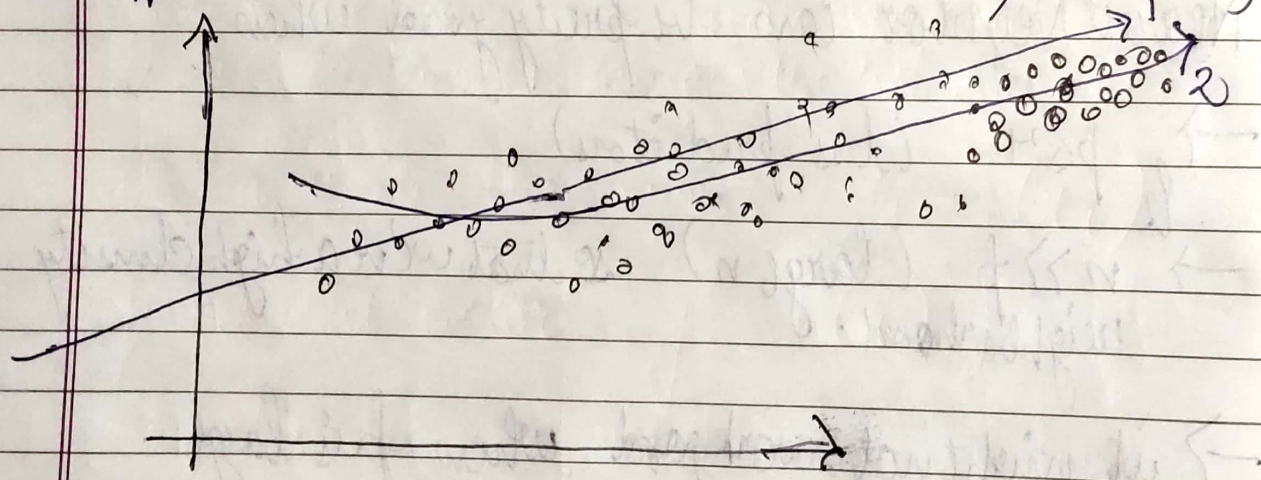
The linear model is a good example of parametric model.

$$f(x) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$$

→ A linear model is specified in terms of "p+1" parameters.

→ We estimate the parameters by fitting the model to the data (training).

→ Although it is almost never correct, a linear model serves as a good & interpretable approximation to the true $f(x) / E(Y|X=x)$



1. Represents linear models

$$f(x) = \beta_0 + \beta_1 X$$

2. Represents Quadratic Model

$$f(x) = \beta_0 + \beta_1 X_1 + \beta_2 X_2^2$$

(It fits slightly better)

* These both are Linear Models but with different parametric forms

Date :
Page :

Parametric Methods involve a two-step method approach:

1. First, is to take an assumption of function form or shape of f.
For example, a linear function.
2. Second step involves training the model & estimating the values of βs.

Non Parametric Methods:

These do not make explicit assumptions about the functional form of f.

Instead they seek to estimate "f" & get as close as possible to it.

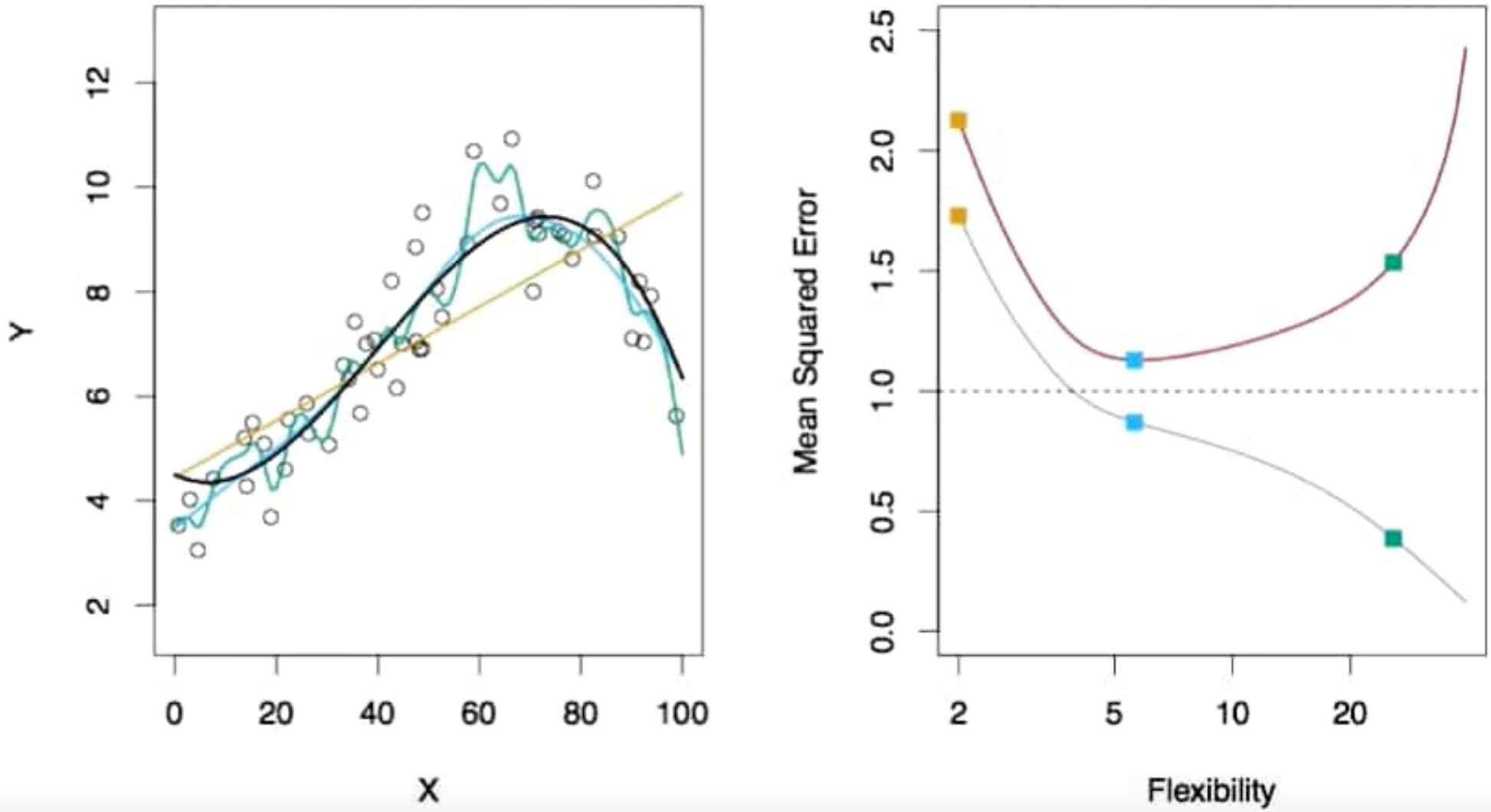
The only disadvantage is it end up overfitting the data.

The Tradeoff Between Prediction Accuracy & Flexibility

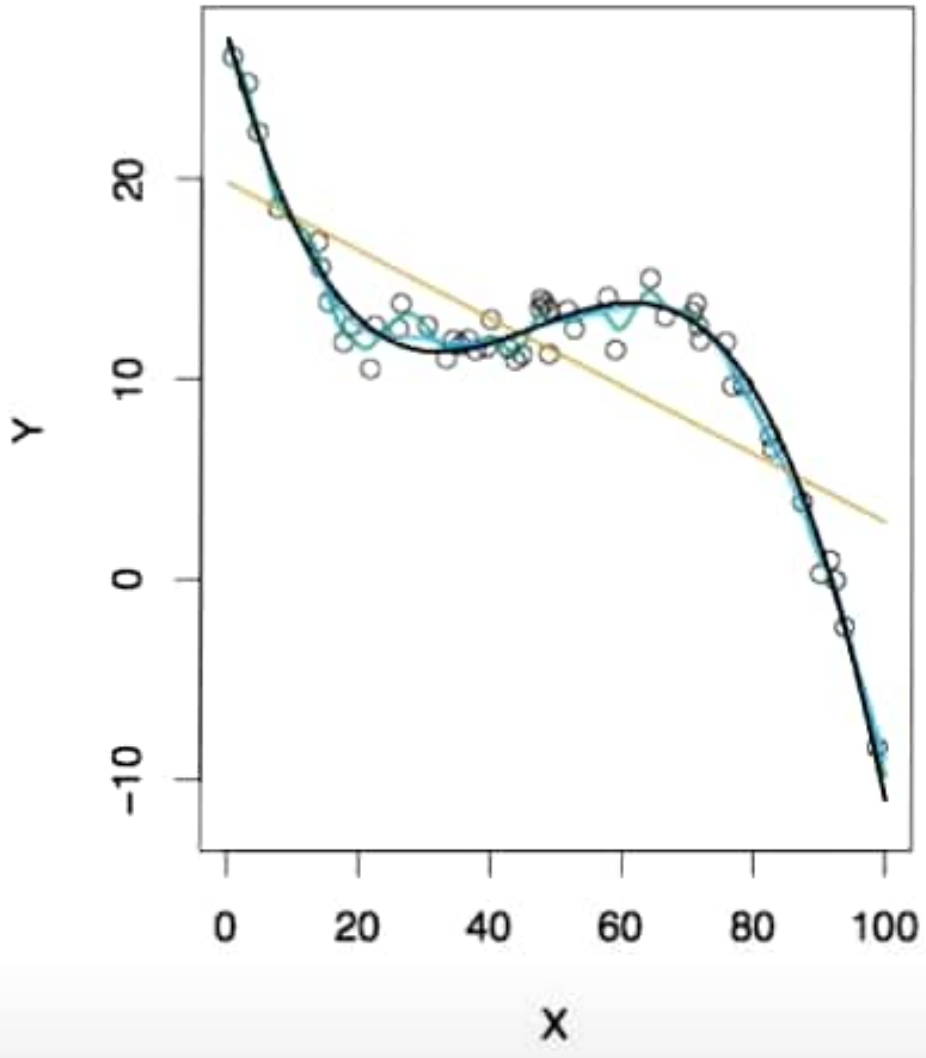
There are multiple methods we can use for predictions some are less flexible or more restrictive.

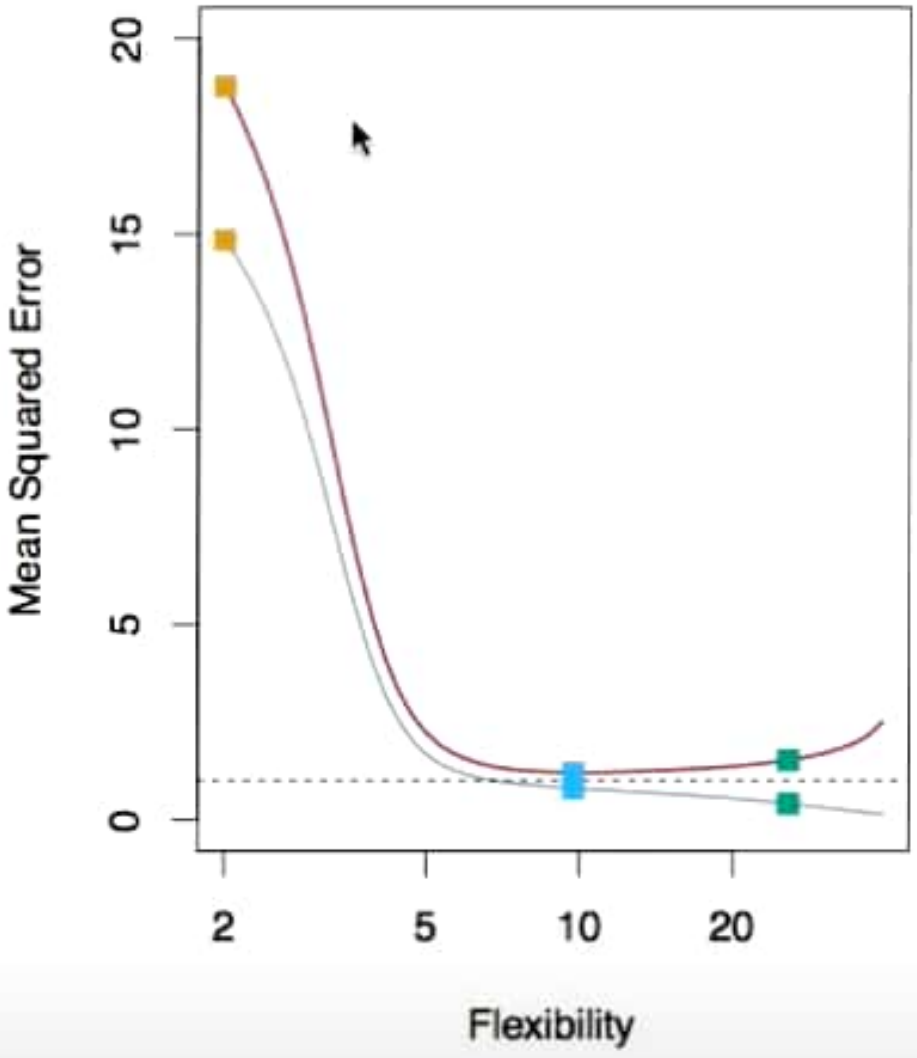
For ex: Linear Regression is very inflexible approach as it can only create linear functions

Other methods such as SVM's or thin plate splines are more flexible.

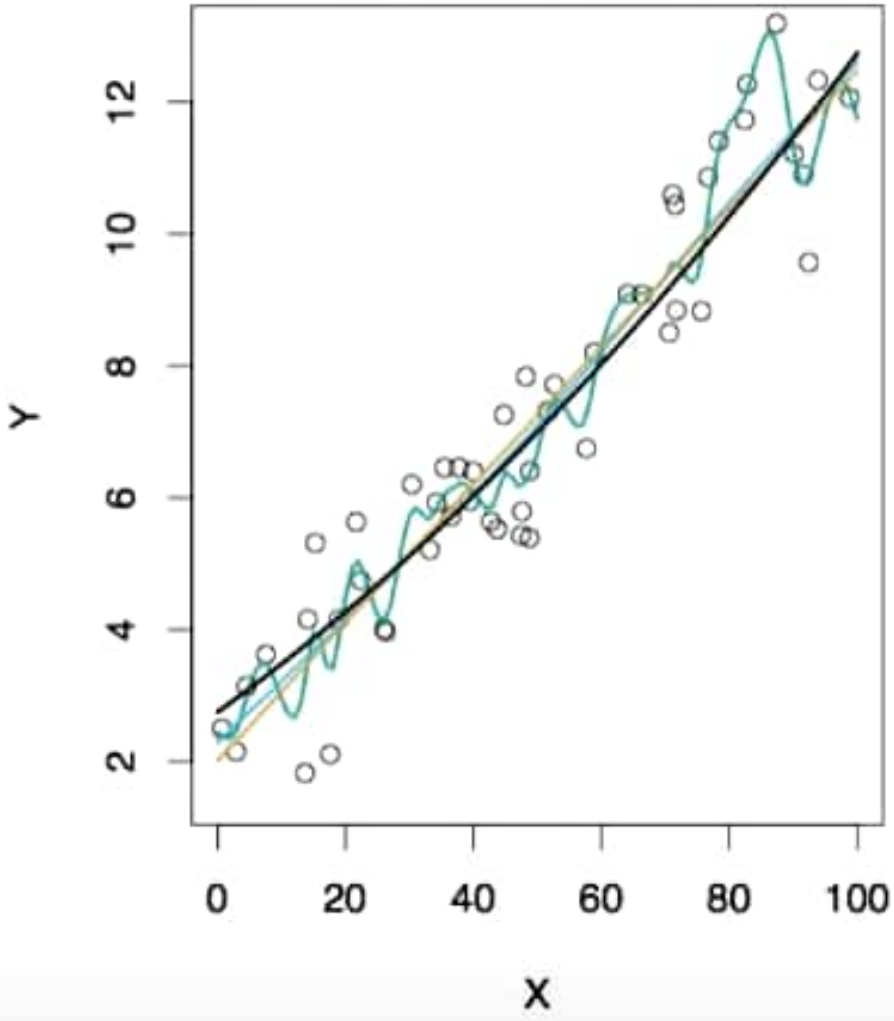


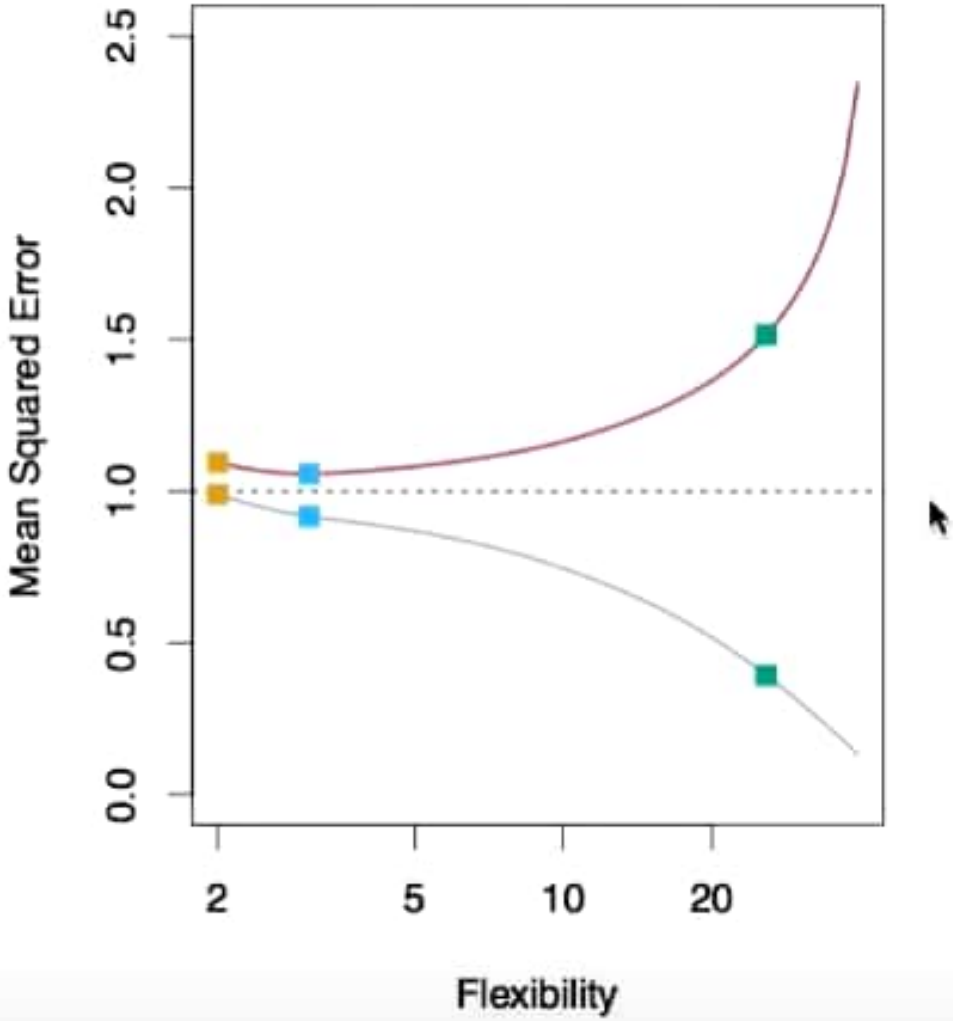
Black curve is truth. Red curve on right is MSE_{Te}, grey curve is MSE_{Tr}. Orange, blue and green curves/squares correspond to fits of different flexibility.





Here the truth is wiggly and the noise is low, so the more flexible fits do the best.





Here the truth is smoother, so the smoother fit and linear model do really well.

So, this question arises, why would we ever choose restrictive models?
There are several reasons:
→ They are more interpretable
→ less prone to overfit

Added a photo above.

Assessing Model Accuracy:

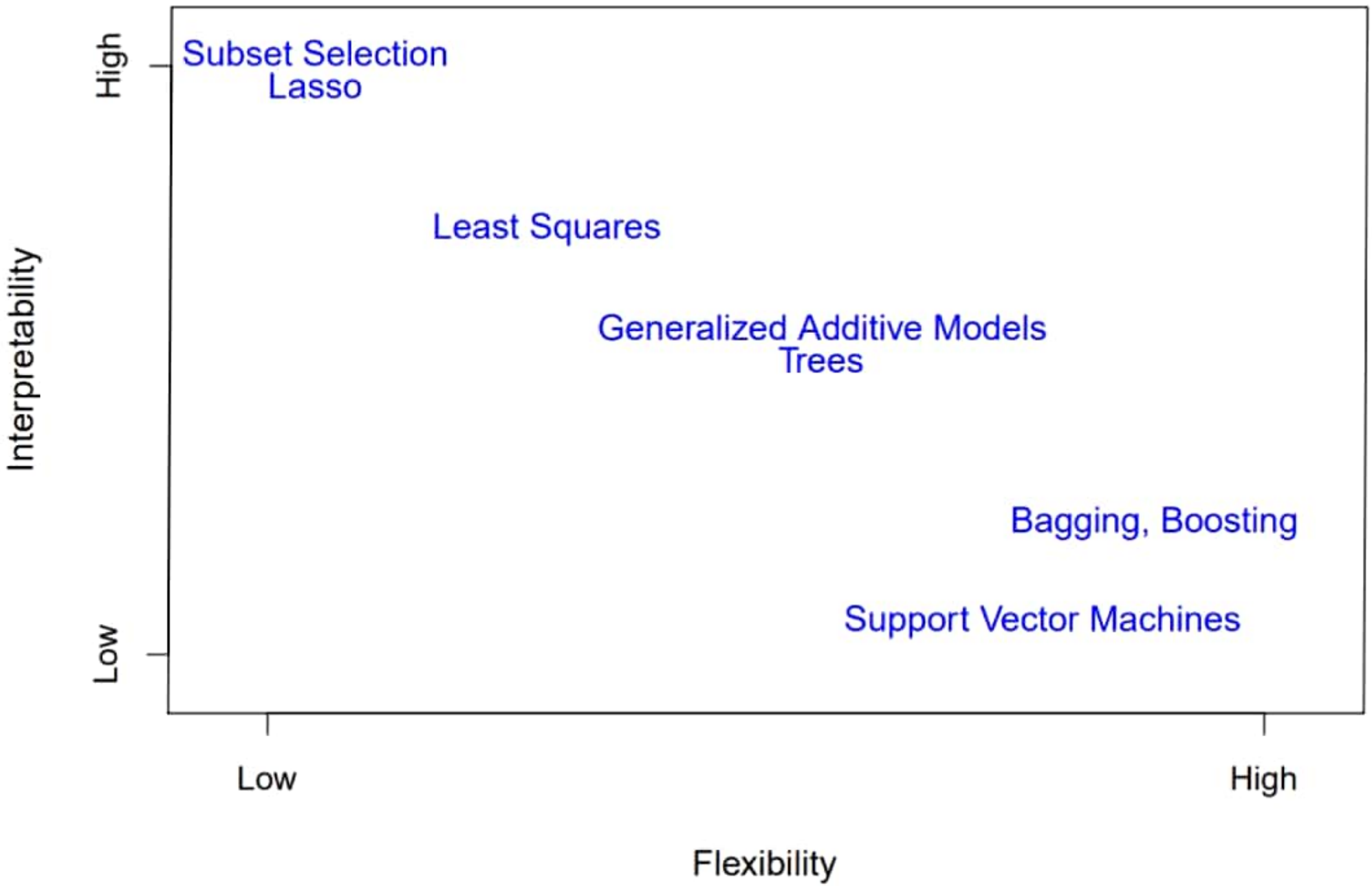
Let us suppose we fit a model to a linear data & that is $\hat{f}(x)$ & we wish to see how it performs.

To do so, we calculate Mean Squarred Error
It is exactly the same as it sounds:
We calculate the error: $y_i - \hat{f}(x_i)$
Square the error: $(y_i - \hat{f}(x_i))^2$
Take it's mean over all the training data.

$$\boxed{\sum_{i=Tr} (y_i - \hat{f}(x_i))^2 / N_{Tr}}$$

But this may be biased toward overfit models
So, we calculate:

$$MSE_{Te} = \sum_{i=Te} (y_i - \hat{f}(x_i))^2 / N_{Te}$$


Subset Selection
Lasso
Least Squares
Generalized Additive Models
Trees
Bagging, Boosting
Support Vector Machines
High
Low
Interpretability
Low
High
Flexibility

where, N_{TE} & N_{TR} is the number of samples of Training & Testing Data respectively.

Bias-Variance Tradeoff:

Suppose we've to fit a model $\hat{f}(x)$ to some training data T_R, & let (x_0, y_0) are Test samples drawn from population & true model is

$$Y = f(X) + \epsilon$$

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = Var\left(\hat{f}(x_0)\right) + \left[Bias\left(\hat{f}(x_0)\right)\right]^2 + Var(\epsilon)$$

where $Bias\left(\hat{f}(x_0)\right) = E\left[\hat{f}(x_0)\right] - f(x_0)$

So, as flexibility of $\hat{f}$ increases Bias Decrease (Overfitting) & variance increases (as the new testing data will have increased variance i.e. chances of misclassification due to overfitting). In high flexible data a little change in x can change f very much hence variance of f increases.

Classification Problem:

Here Y is Qualitative / Categorical not a continious variable say $C = (spam, ham)$ or $C = \{0, 1, \ldots, 9\}$ digits.

Our goals are:

→ Building a classifier to predict classes for unlabelled data.
→ Assess uncertainity in each classifications.
→ Understand the role of all the predictors.

The concepts like "bias variance tradeoff" & "Flexibility Interpretablity Tradeoff" are transferable here.

Here, the model is Assessed Using:

$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$

which returns 1 when $y_i \neq \hat{y}_i$
& 0 when $y_i = \hat{y}_i$

So, it gives the proportion of Training Error.

Naive Bayes Classifier:

$$P_K(x) = Pr(Y = K \mid X = x)$$
where $K = 1, 2, \ldots, K$

This is called a conditional probability at a given x.

The Bayes Classifier at x is:

$$C(x) = j \text{ if } P_j(x) = \max(P_1(x), P_2(x), \ldots, P_K(x))$$

where, Bayes Error Rate is given by

$$1 - E\left(\max_j Pr(Y = j \mid X)\right)$$

Date :
Page :

Nearest Neighbor:

The nearest neighbor averaging can be done as before. This also breaks down as p dimensions grow.

→ Model is accessed using $\hat{f}(x_0)$

$$= E\left(I\left[y_i \neq \hat{y}_i\right]\right)$$

$$= \frac{1}{N} \sum I\left[y_i \neq \hat{y}_i\right]$$

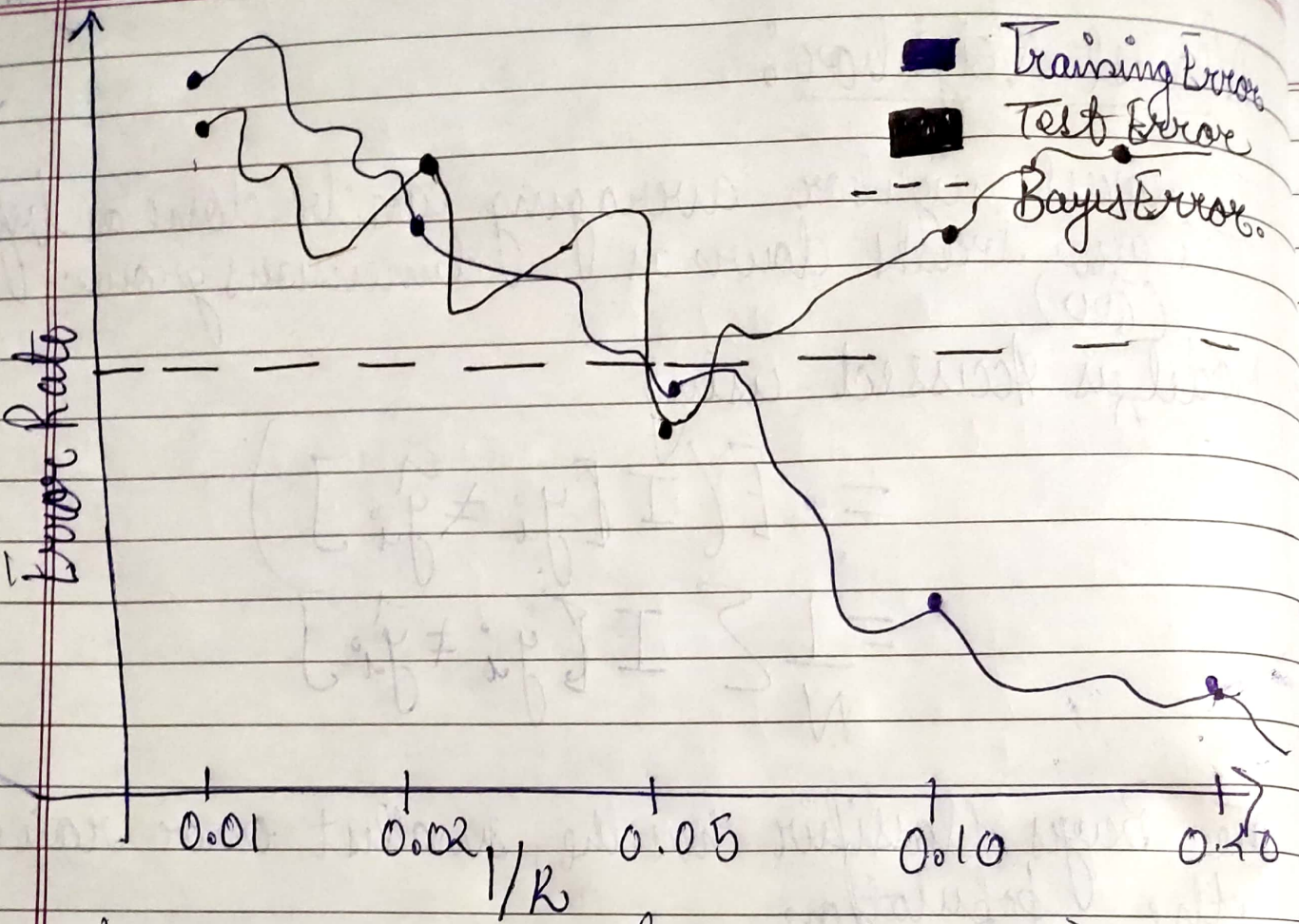
→ The Bayes Classifier has the smallest error rate in the population

→ SVM builds structured model for $\hat{f}(x)$.

→ also models like Logistic Regression, boosting & bagging.

In K-nearest neighbors we calculate the "K" smallest distant point from the data say (x_0) selects $(x_{10}, x_{21}, x_{32}, x_{03} \ldots x_{0K})$ an averages & takes the output with high probabilities.

	I/P	O/P
Say 5 neighbors are:	{0, 0, 0, 1, 1} =	0
	{0, 0, 0, 0, 1} =	0
	{1, 1, 1, 0, 0} =	1
	{1, 1, 0, 0, 0} =	0
	{1, 1, 1, 0, 1} =	1



We can see test error rate increase after some time again
& obviously nothing does better job than naive bayes theoretically.