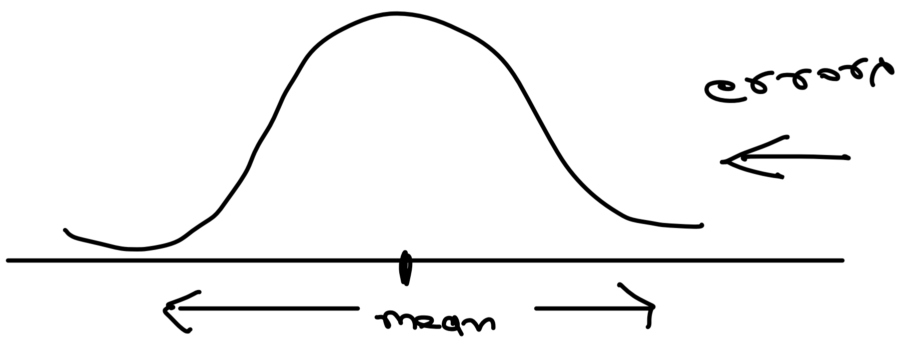
* Agenda and Recap
    * Normality of Residuals
    * Homoscedasticity
    * Auto-collinearity

  3

* Gradient Descent Variants
* Polynomial Regression
* Generalization and Occam's razor
* Underfitting and overfitting
* Bias Variance Tradeoff

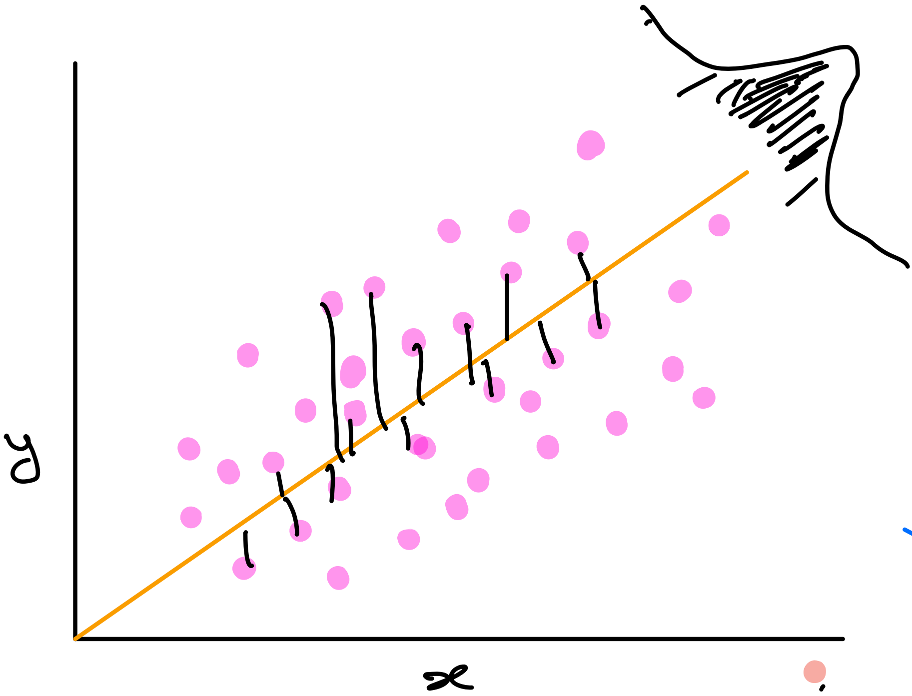
Assumption 3: Errors are Normally distributed

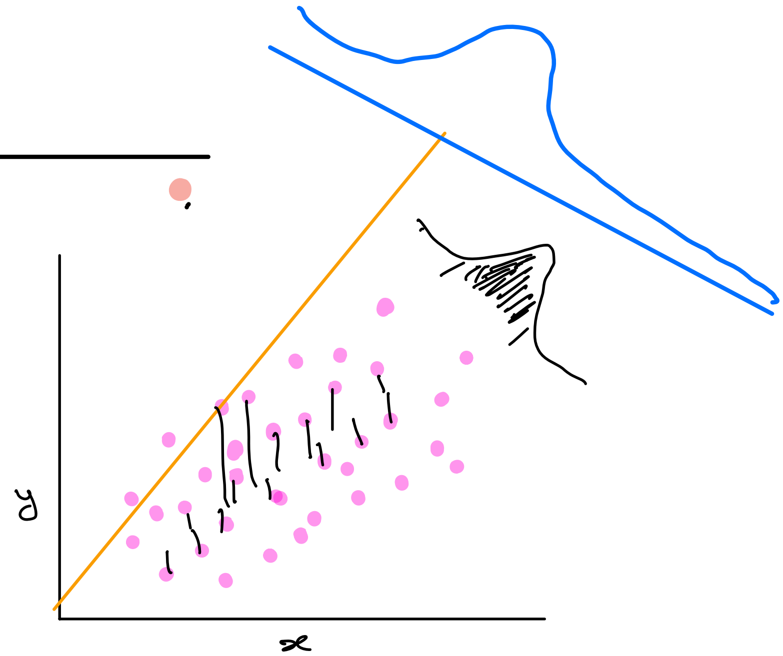


Step 1: Build Model

Step 2: Calculate Error

Step 3: Plot Errors with Histogram

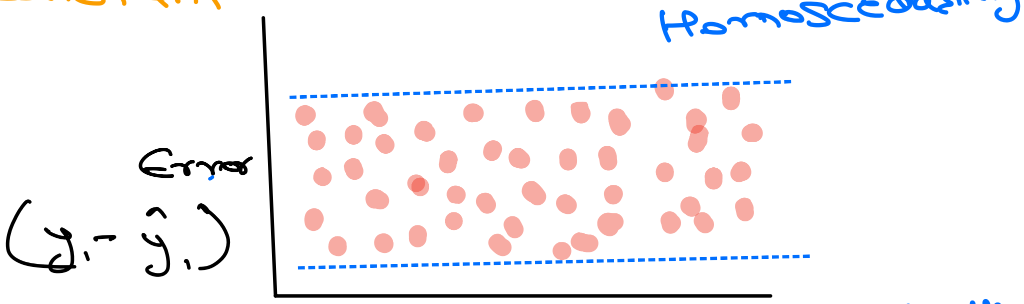


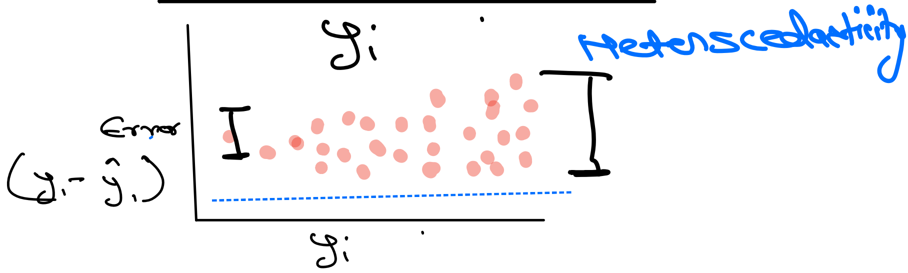


Assumption 4: Heteroskedasticity should not exist

⇒ Variance of residuals (ϵ_i) vs predictions ($\hat{y}_i$) should be constant

$$\boxed{y_i \text{ vs } \epsilon_i}$$





* Goldfeld Quandt Test

⇒ Null Hypothesis: Dataset has Homoscedasticity

* If p value ≤ significant level Threshold
  ⇒ reject Null Hypothesis

* To mitigate:

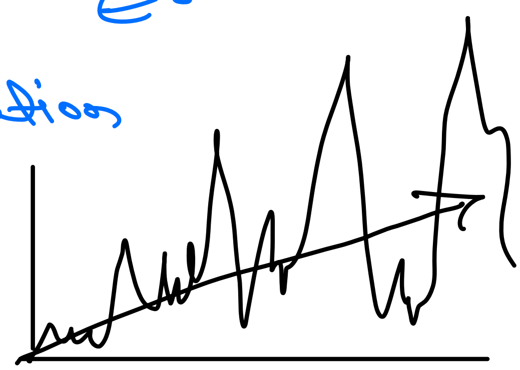
⇒ Remove outlier

⇒ Perform Non-Linear Transformation such as box-cox

(Correlate between Errors)

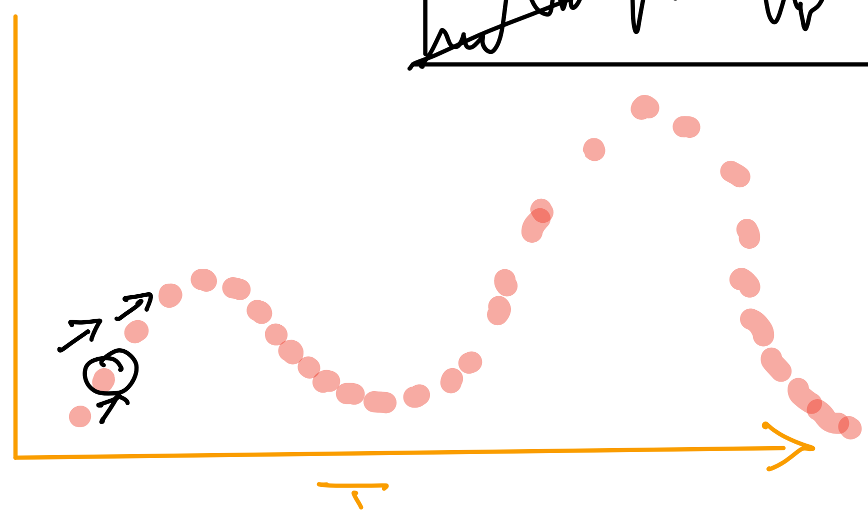
Assumption 5: No Autocorrelation

↓ Self



ϵ_i vs T

⇒ Time-series Model



Variants of SGD

(Just 1 update per iteration)

$$w_j^{new} = w_j^{old} - \eta \frac{\partial L}{\partial w_j^{old}}$$

1000 ⇒ 1000 updates
iter

$$\frac{2}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i) \times x_{ij}$$

For One Single iteration/update

1 Million

⇒ 1 million Errors and Gradients

⇒ Update Weights

1 million

k ⇒ between 1 and N

let's $k = 256$

$$\sum_{i=1}^{\text{Batches}} \frac{2}{k} \sum_{i=1}^{k} (\hat{y}_i - y_i) \times x_{ij}$$

$$w_j^{new} = w_j^{old} - \eta \frac{\partial L}{\partial w_j^{old}}$$

Batches ⇒ $\frac{1 \text{ million}}{256}$ Updates ←

(updates per iteration)

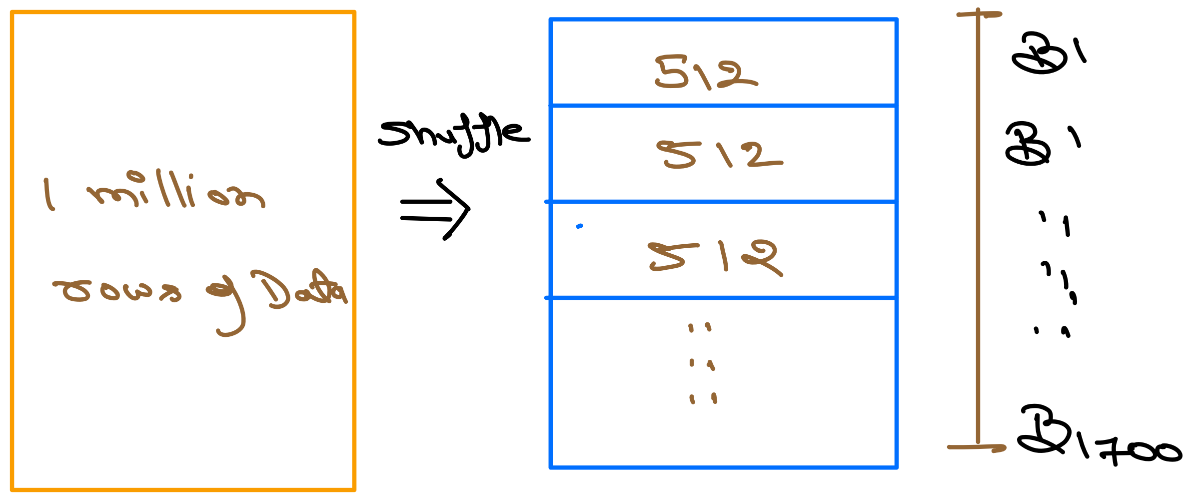
1000 iters → 1000 × Batches

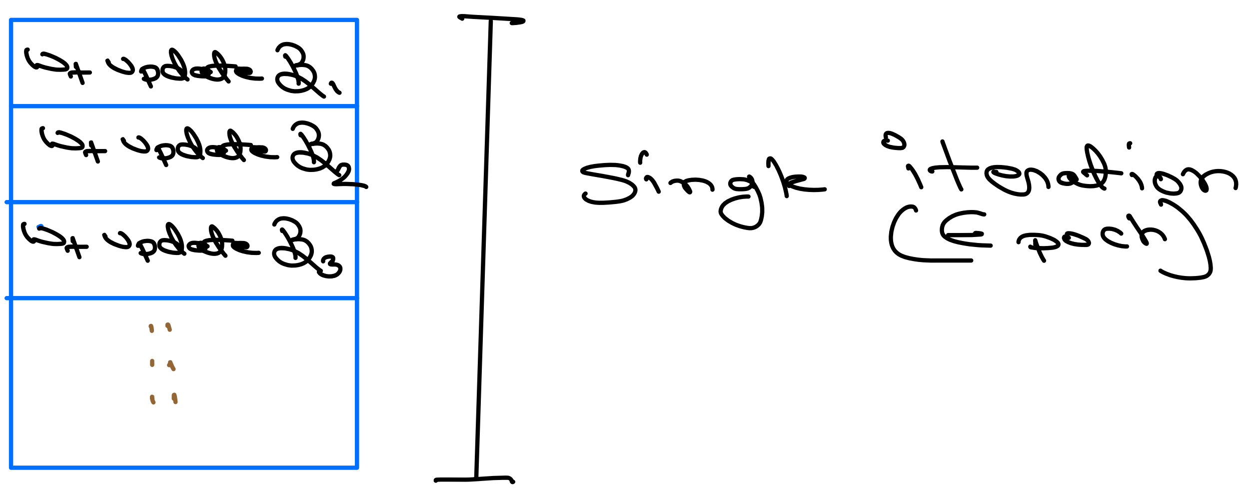
k → 1 (Stochastic Gradient Descent (SGD))

→ N (Batch GD)

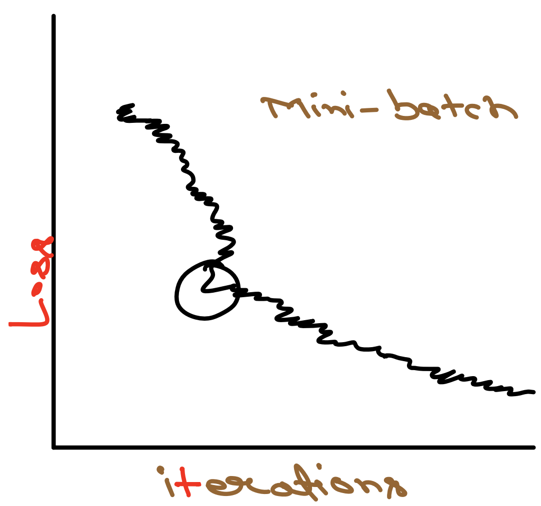
→ (1, N) (Mini-Batch GD) MGD

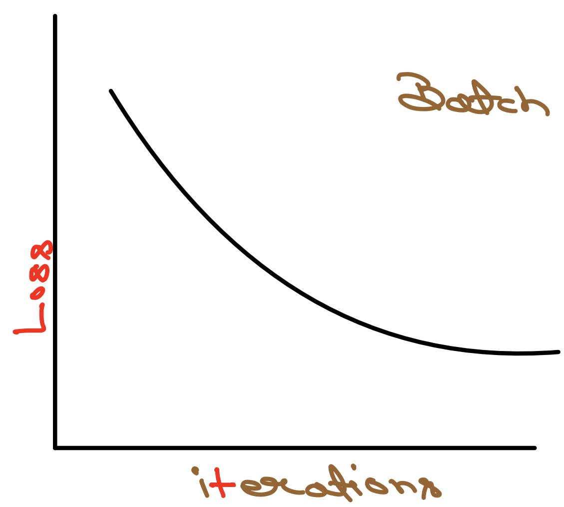
(Weight update in Single Epoch)





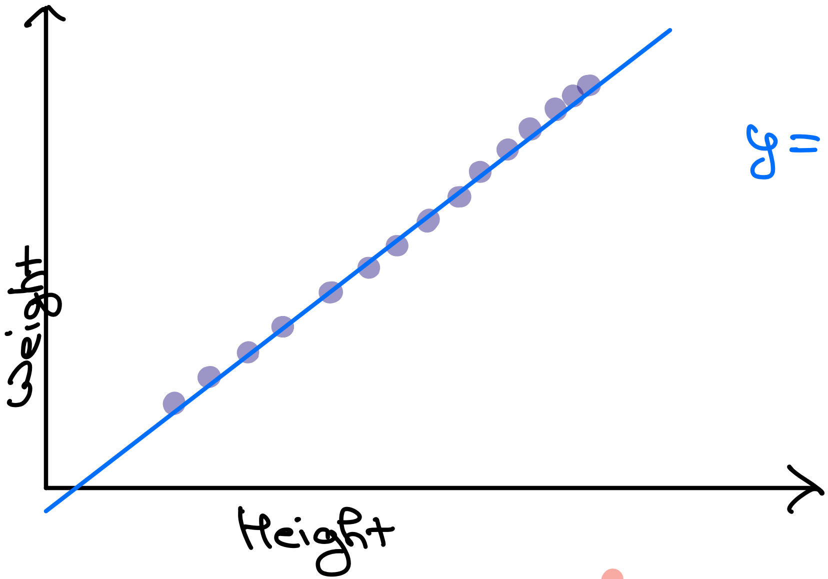
Comparison: Minibatch vs Batch



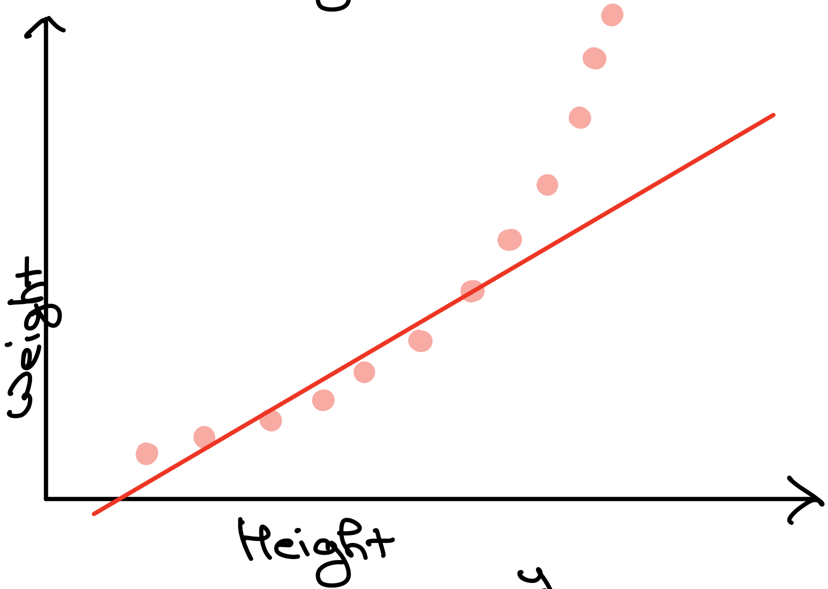


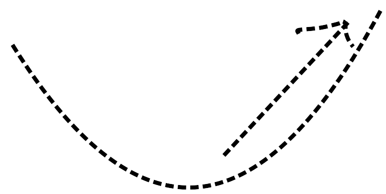
Polynomial Regression

John



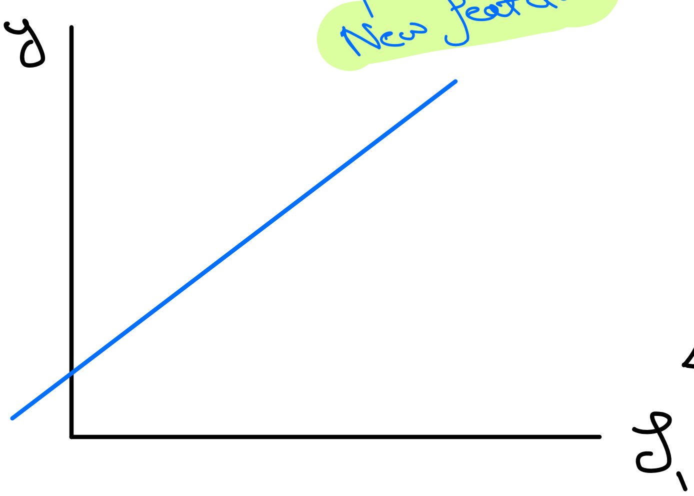
$$y = \omega_1 x_1 + \omega_0$$

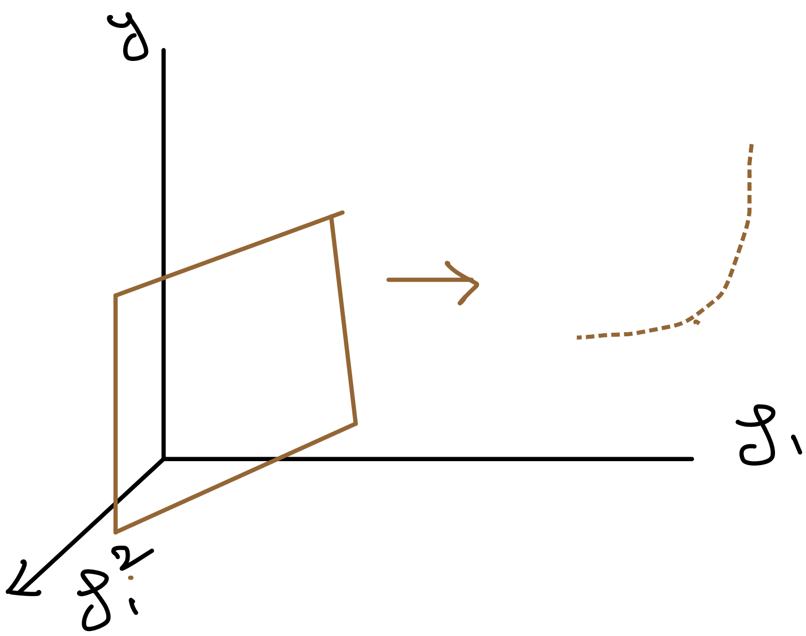




$$y = \omega_1 f_1 + \omega_2 f_1^2 + \omega_0$$

New feature





$f_2 \Rightarrow f_2^2$

$f_1 \times f_2$

* How Does this Look

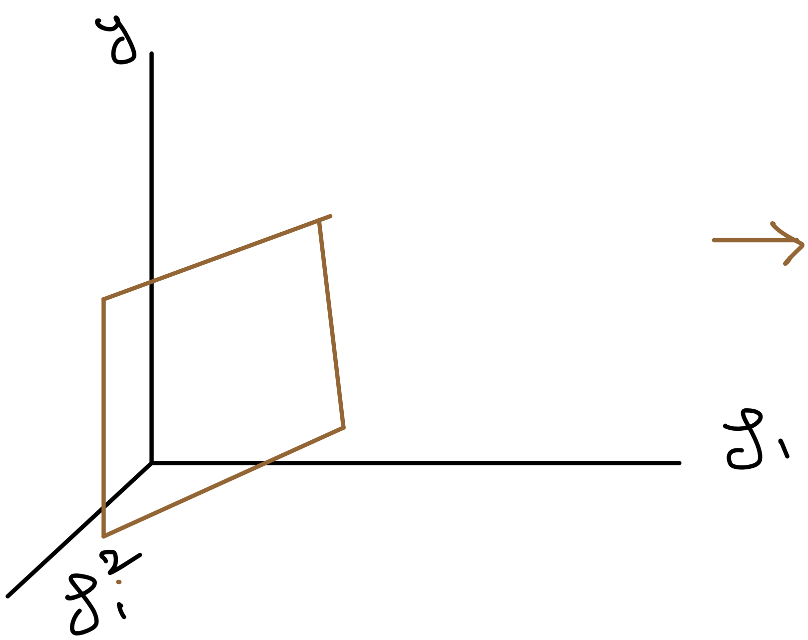
-> In data

x	y
x_{11}	y_1
x_{21}	y_2
x_{31}	y_3
x_{41}	y_4
x_{n1}	y_n

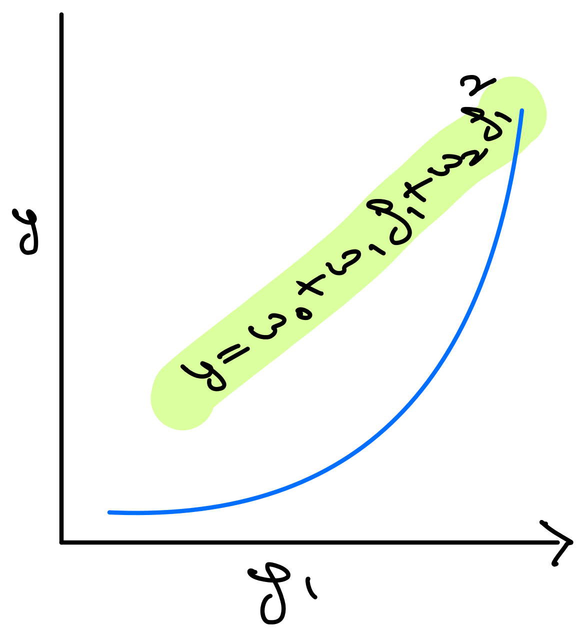
$\Rightarrow$

f_1^2	f_1	
x^2	x_1	y
x_{11}^2	x_{11}	y_1
x_{21}^2	x_{21}	y_2
x_{31}^2	x_{31}	y_3
⋮	x_{41}	y_4
x_{n1}^2	x_{n1}	y_n

-> In plot.



$\rightarrow$

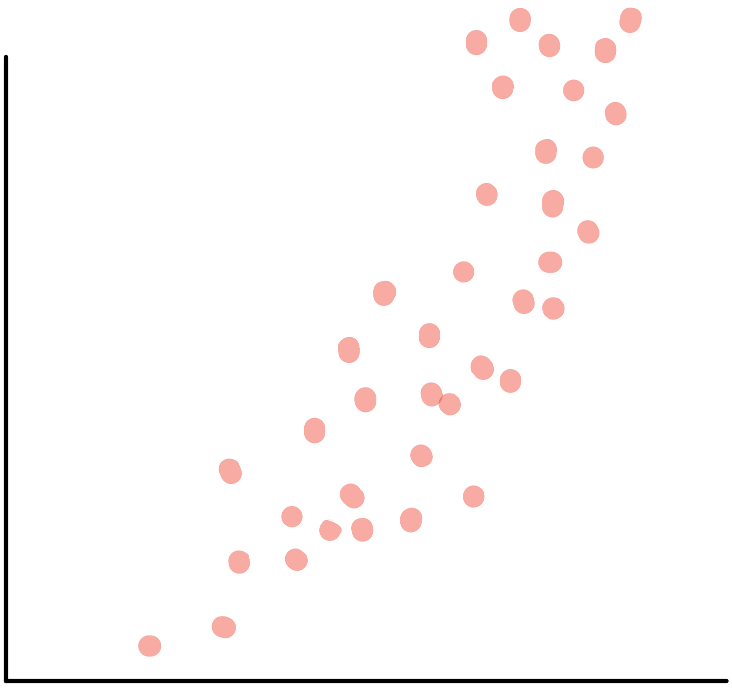


* What about Multi-collinearity?

✗ $f_2 = \alpha f_1 + \beta$ ← Linear Relationship

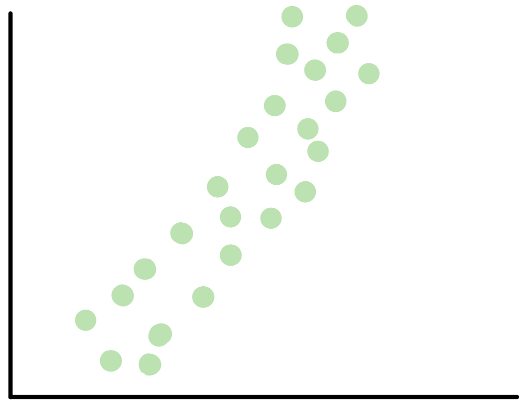
✓ $f_2 = f_1 \times f_1$ ← Non Linear

Generalization and Occams Razor

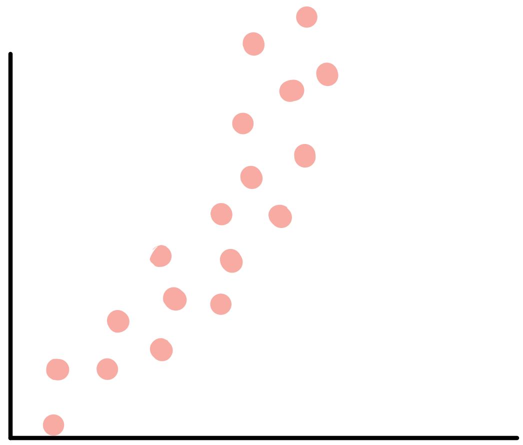


full dataset

→



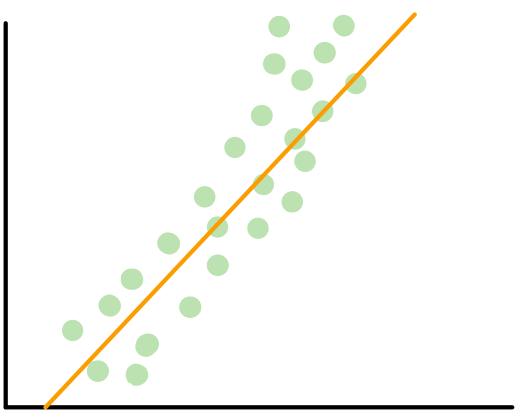
Train Set



Test-set

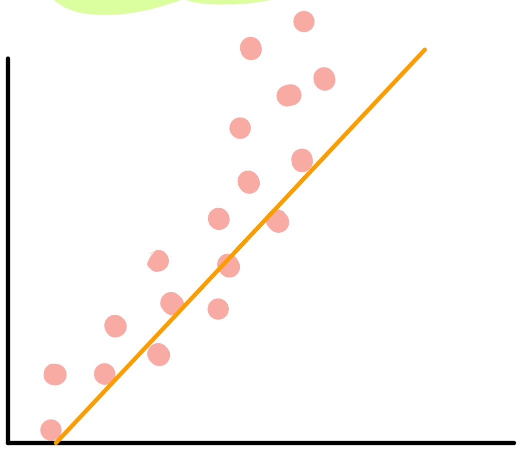
Model-1

Underfitting



Train Set

$r2$-score = 0.65

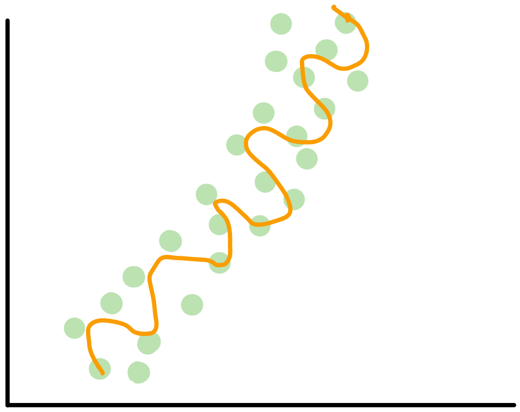


Test-Set

$r2$-score = 0.60

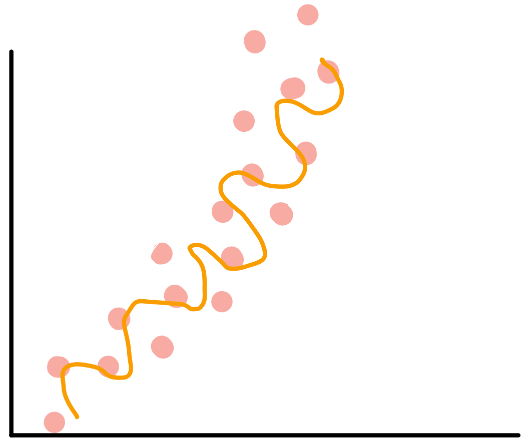
Model 2

Overfitting



Train Set

r2-score = 0.90

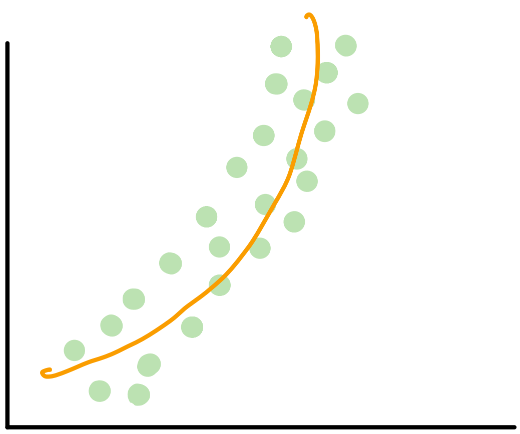


Test-Set

r2-score = 0.85

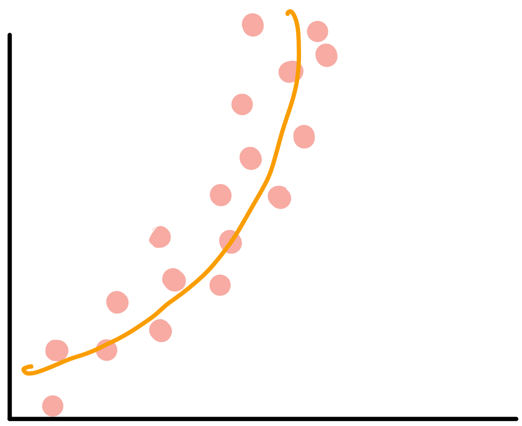
Model 3

Good Fit



Train Set

r2-score = 0.85



Test-Set

r2-score = 0.84

⇒ M3

Simplest

Model	Training	Testing
Underfit	bad	bad
Perfect	Decent	Decent
Overfit	Excellent	Decent or bad

Complex

Principles for picking the Best Model

Generalization

1) Pick model which performs the best on Unseen Data (Generalizes)

Occam's Razor

2) If There are many Solutions pick the Simplest One.

M2 $\simeq$ M3
(complex) (simple)

* Higher the degree More Complex the Model

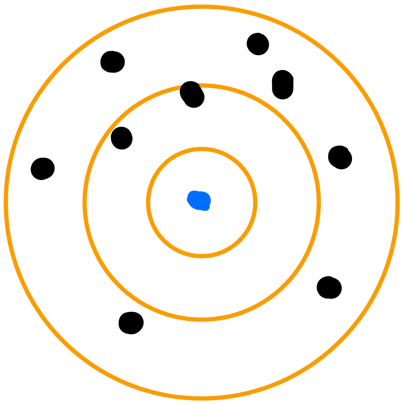
Bias and Variance Trade-off

~~wor~~ ~~SD~~ Variance

(Two types of Error)

Vivek is preparing for Olympic archery Competition

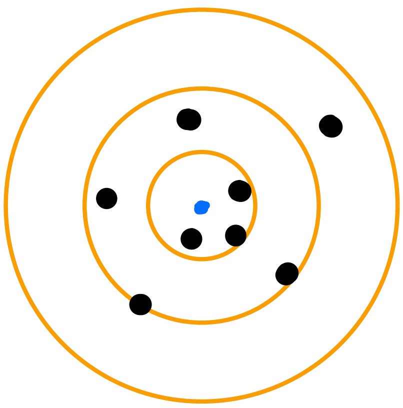
Case-1



⇒ High Bias ⇒ The shots are far away from Bull's eye

⇒ High Variance ⇒ Not Consistent (all over the board)

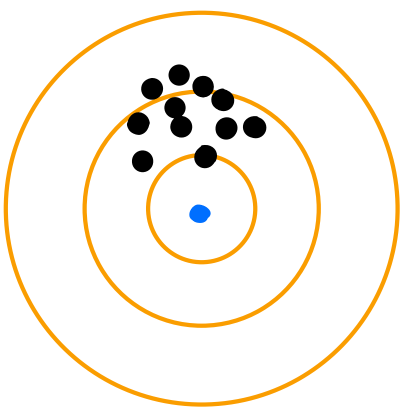
Case-2



Low Bias

High Variance

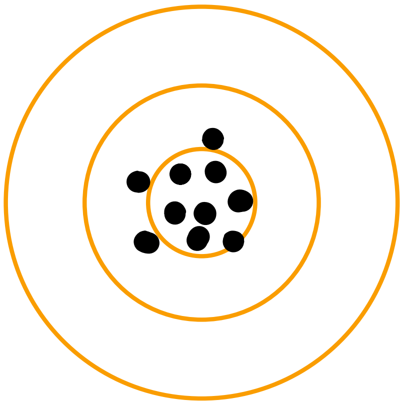
Case 3 :



High Bias

Low Variance (Consistent)

Case - 4



Low Bias
Low Variance

* Conclusion:

1) High bias leads to high Errors
2) High Variance leads to high Errors

Bias vs Variance Tradeoff