Advanced Machine learning Mastering Course

Introduced by

George Samuel

Master in computer science
Cairo University

Innovisionray.com

2024

Agenda

Supervised Learning-Regression

Normal Equation

We use the normal equation to calculate the optimal weights which Leeds us to the global minimum in one step

Polynomial Regression

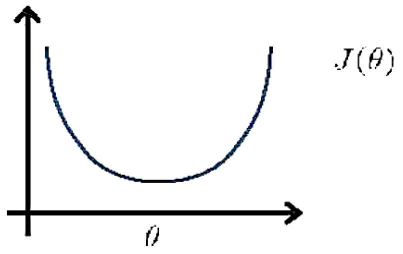
$$c_0+c_1x+c_2x^2+c_3x^3=y_1$$

X	Y
X_1	y_1
X_2	y_2
.	.
.	.
.	.
.	.
X_n	y_n

$$\underbrace{\begin{vmatrix} \underline{1\ X_1\ X^2{}_1 X^3{}_1} \\ 1\ X_1\ X^2{}_1 X^3{}_1 \\ 1\ X_1\ X^2{}_1 X^3{}_1 \\ . \\ . \\ . \\ . \\ 1\ X_n\ X^2{}_n X^3{}_n \end{vmatrix}}_{X} \ \underbrace{\begin{vmatrix} C0 \\ C1 \\ C2 \\ C3 \end{vmatrix}}_{W} \approx \underbrace{\begin{vmatrix} Y1 \\ Y2 \\ Y3 \\ . \\ . \\ . \\ . \\ . \\ Yn \end{vmatrix}}_{Y}$$

$$\hat{\theta} = (X^T \cdot X)^{-1} \cdot X^T \cdot y$$

Gradient Descent



Normal equation: Method to solve for θ analytically.

Supervised Learning-Regression

Normal Equation

$$\hat{\theta} = (X^T \cdot X)^{-1} \cdot X^T \cdot y$$

example

$$X = \begin{bmatrix} 1 & 1 & 2 & 3 \\ 1 & 0 & 4 & 5 \\ 1 & 1 & 0 & 6 \end{bmatrix} \qquad y = \begin{bmatrix} 5 \\ 20 \\ 15 \end{bmatrix}$$

$$X = \begin{bmatrix} \boxed{\begin{matrix} 1 \\ 0 \\ 1 \end{matrix}} & \boxed{\begin{matrix} 2 \\ 4 \\ 0 \end{matrix}} & \boxed{\begin{matrix} 3 \\ 5 \\ 6 \end{matrix}} \end{bmatrix} \qquad X.t = \begin{bmatrix} \boxed{1 \quad 0 \quad 1} \\ \boxed{2 \quad 4 \quad 0} \\ 3 \quad 5 \quad 6 \end{bmatrix}$$

$$X.t * X = \begin{bmatrix} \boxed{2} & \boxed{2} & \boxed{9} \\ \boxed{2} & 20 & 26 \\ 9 & 26 & 70 \end{bmatrix}$$

Normal Equation $\hat{\theta} = (X^T \cdot X)^{-1} \cdot X^T \cdot y$

$$X = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 1 & 0 & 6 \end{bmatrix}$$

$$(X*X.t)^{-1} = \frac{1}{||A||} * \text{adj}(A)$$

$||A|| = 1(4*6) – 2(0-5)+3(0-4) = 22$

calculate the adj by using detirmnant for each number in matrix A

adj = $\begin{bmatrix} 24 & -5 & -4 \\ 12 & 3 & -2 \\ -2 & 5 & 4 \end{bmatrix}$ put the sign = $\begin{bmatrix} 24 & 5 & -4 \\ -12 & 3 & 2 \\ -2 & -5 & 4 \end{bmatrix}$ transpose = $\begin{bmatrix} 24 & -12 & -2 \\ 5 & 3 & -5 \\ -4 & 2 & 4 \end{bmatrix}$

$$(X*X.t)^{-1} = \frac{1}{||A||} * \text{adj}(A)$$

$$\frac{1}{22} * \begin{bmatrix} 24 & -12 & -2 \\ 5 & 3 & -5 \\ -4 & 2 & 4 \end{bmatrix} = \begin{bmatrix} \frac{24}{22} & \frac{-12}{22} & \frac{-2}{22} \\ \frac{5}{22} & \frac{3}{22} & \frac{-5}{22} \\ \frac{-4}{22} & \frac{2}{22} & \frac{4}{22} \end{bmatrix}$$

Supervised Learning-Regression

$$X.T * y = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 4 & 0 \\ 3 & 5 & 6 \end{bmatrix} * \begin{bmatrix} 5 \\ 20 \\ 15 \end{bmatrix} = \begin{bmatrix} 20 \\ 90 \\ 205 \end{bmatrix}$$

$$(X*X.t)^{-1} X.T * y = \begin{bmatrix} \frac{24}{22} & \frac{-12}{22} & \frac{-2}{22} \\ \frac{-5}{22} & \frac{3}{22} & \frac{-5}{22} \\ \frac{-4}{22} & \frac{2}{22} & \frac{4}{22} \end{bmatrix} * \begin{bmatrix} 20 \\ 90 \\ 205 \end{bmatrix}$$

$$\begin{bmatrix} -45.909 \\ -29.772 \\ 41.8181 \end{bmatrix}$$

$$c_0+c_1x+c_2x^2+c_3x^3 = y_1$$

$$c_0+45.909x+-29.772x^2+41.818x^3 = y_1$$

Supervised Learning-Regression

Gradient Descent

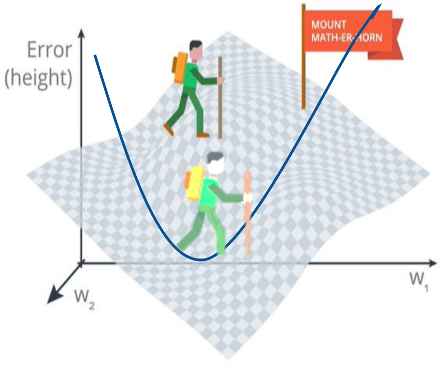
$x^2+-2x+2=0$

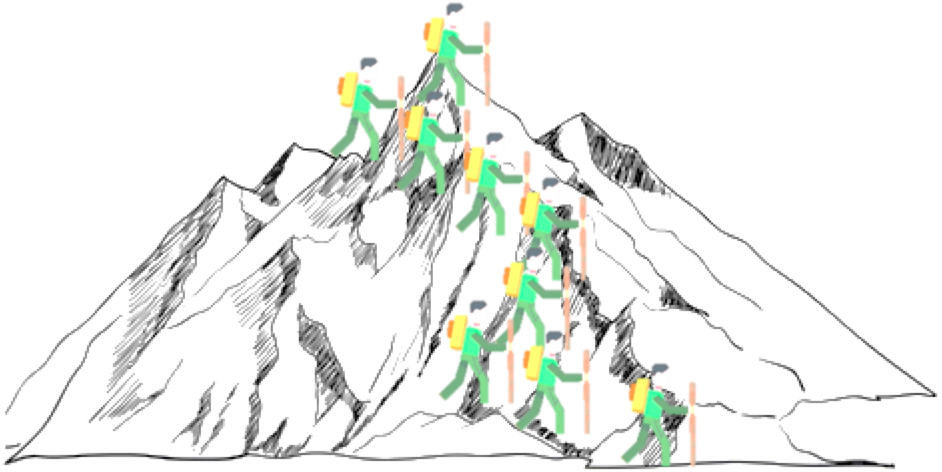
$2x-2=0$

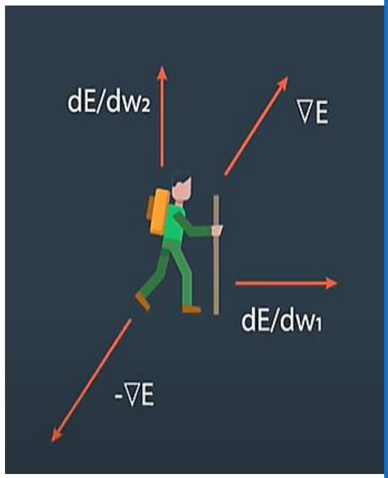
$2x=2$

$2x=2$

$X=1$







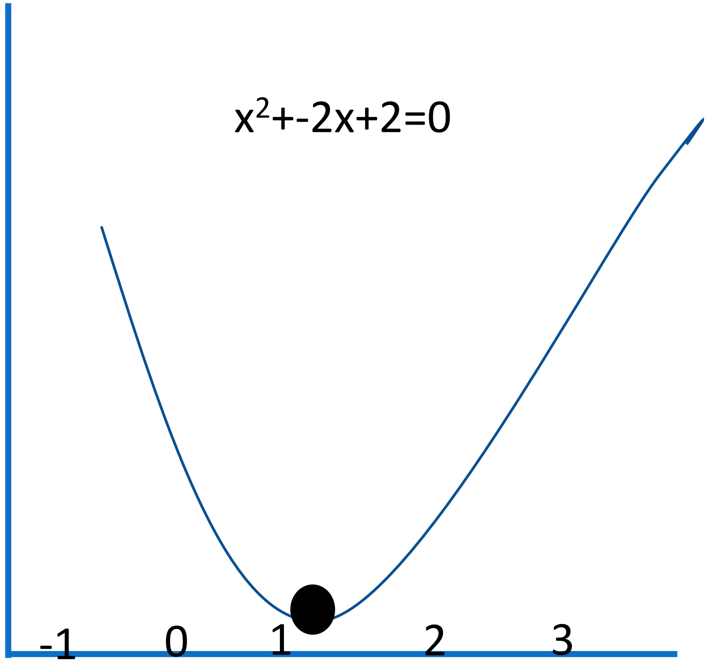
$x^2-14x=0$

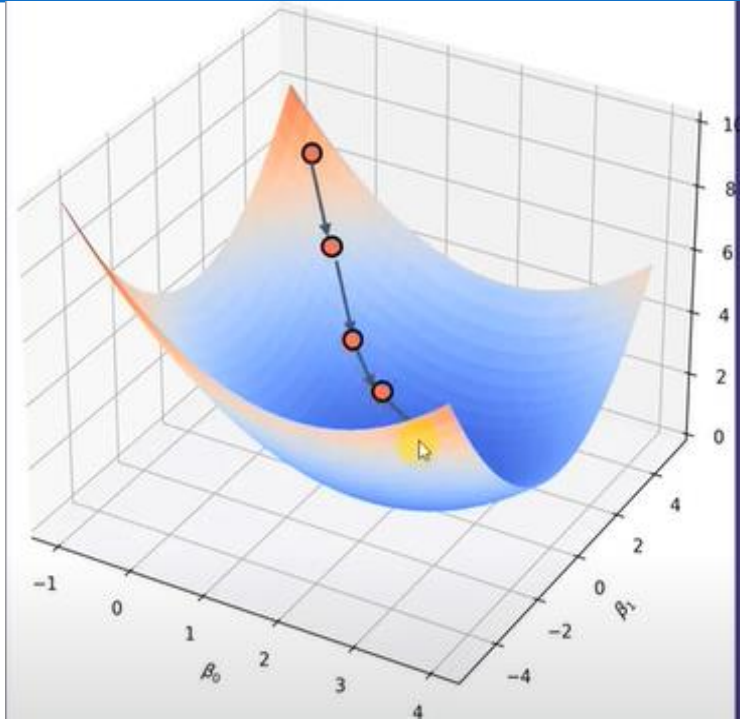
$2x-14=0$

$2x=14$

$x=14/2$

$X=7$





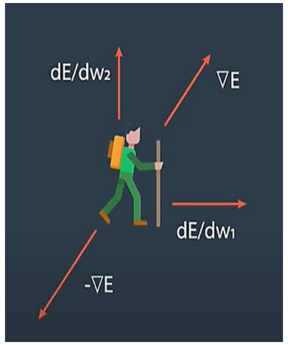
Supervised Learning-Regression

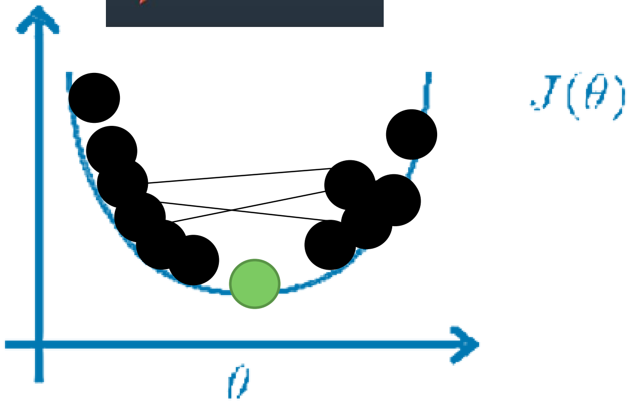
$F(x)=x^3-6x^2+9x+15$

$3x^2-12x+9=0$

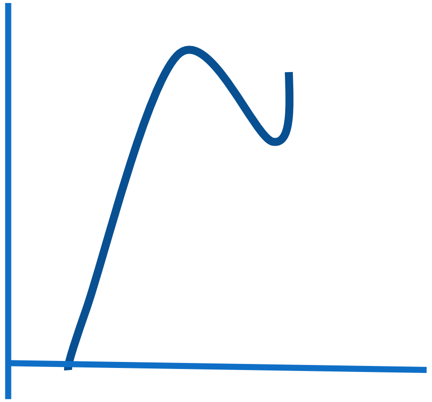
Here we can find max not min

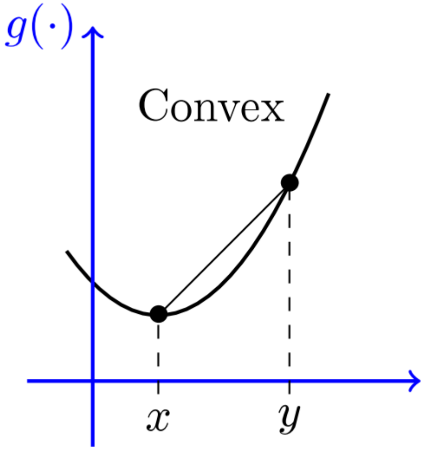
Overshot



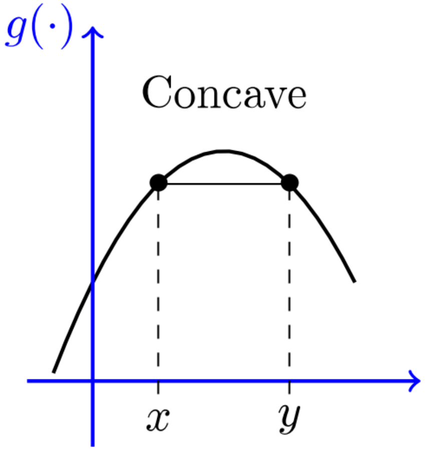


$F(x)=x^3-6x^2+9x+15$



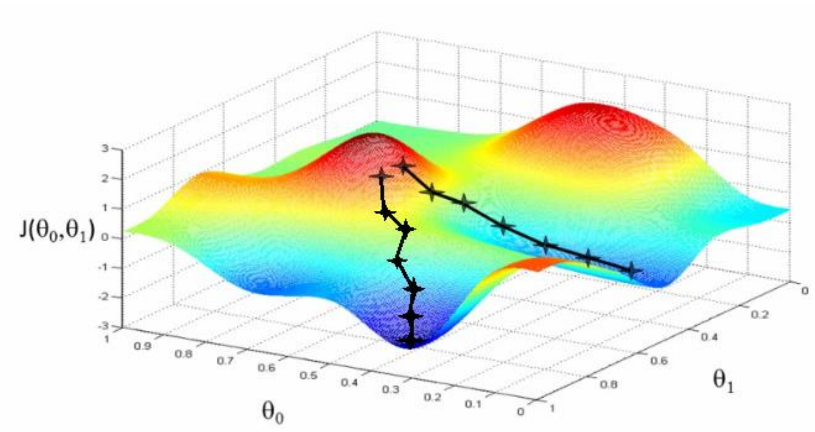


Global minimum



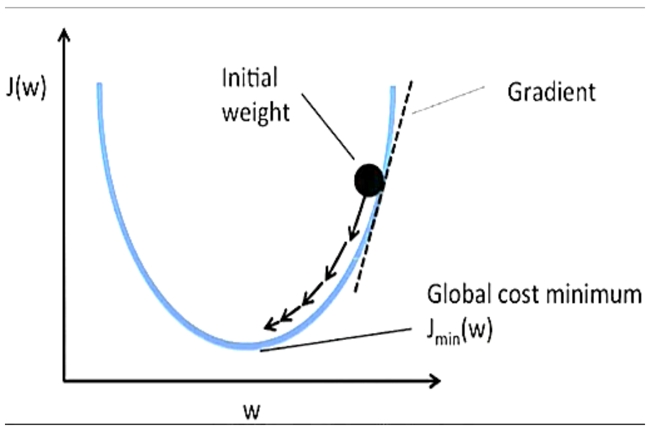
Local minimum

$\hat{y}=\Theta_0+\Theta_1 x$

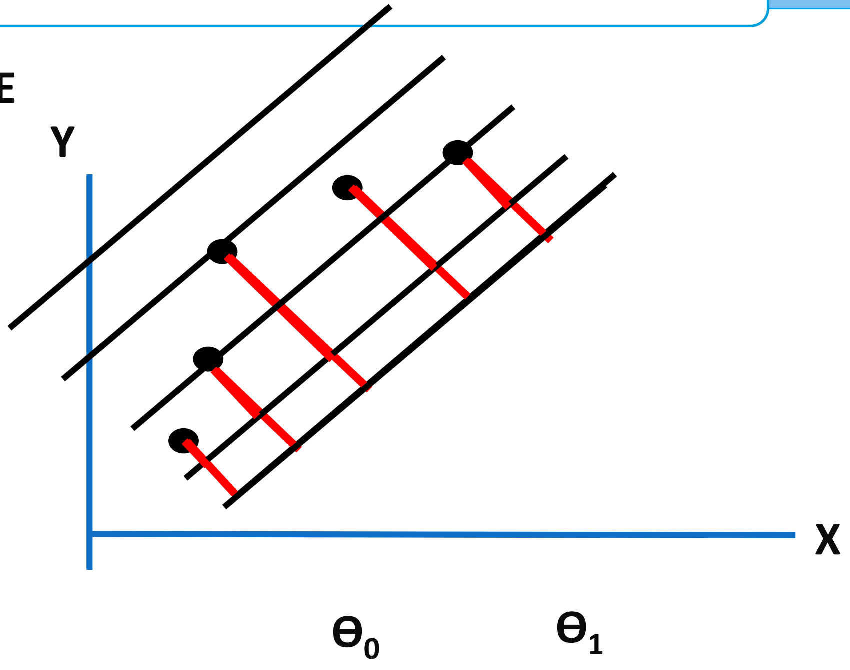


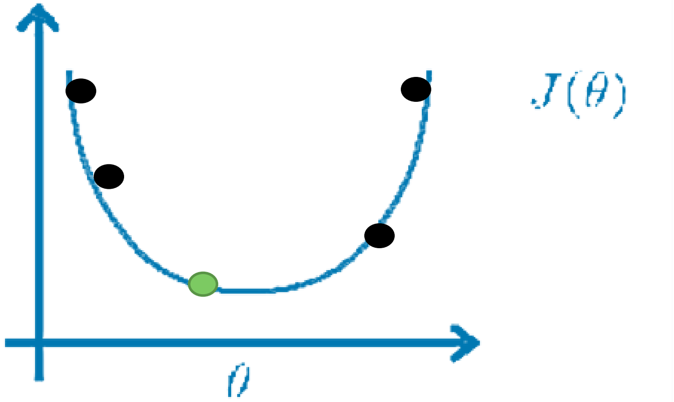
Supervised Learning-Regression

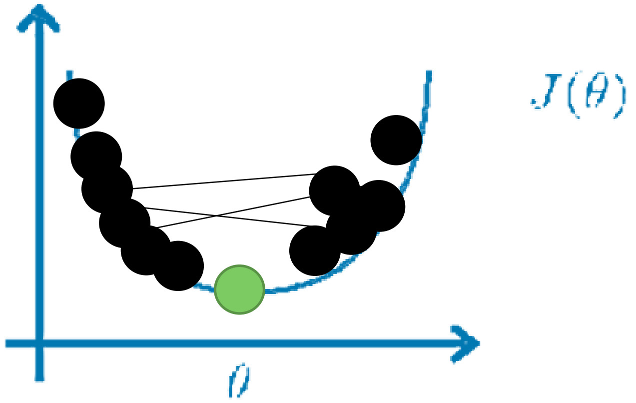
Gradient Descent

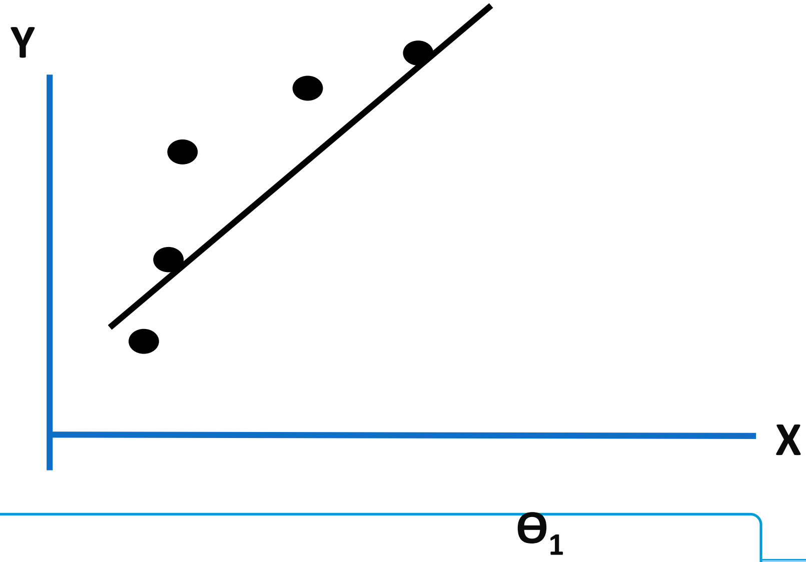


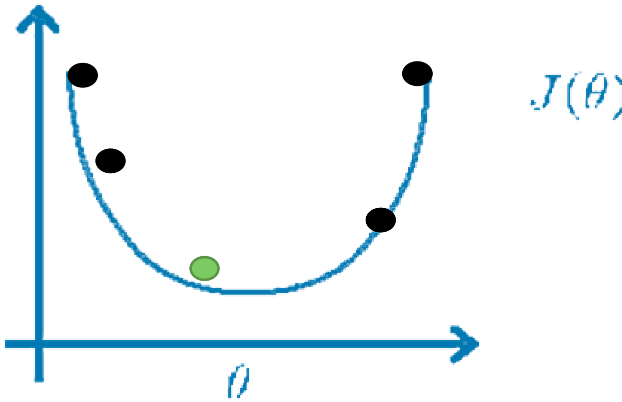
Big MSE











Supervised Learning-Regression

$y̆=\Theta_0+\Theta_1 x$

Cost Function: $J(\theta_0,\theta_1) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$

First choose initial values for Θ_0,Θ_1

Calculates the derivatives for Θ_0,Θ_1

Derivatives:

$$\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)$$

$$\frac{\partial}{\partial\theta_1}J(\theta_0,\theta_1) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)\cdot x^{(i)}$$

Update rules:

$$\theta_0 := \theta_0 - \alpha\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1)$$

$$\theta_1 := \theta_1 - \alpha\frac{\partial}{\partial\theta_1}J(\theta_0,\theta_1)$$

Supervised Learning-Logistic Regression

Regularization: Ridge and Lasso Regression

Revision

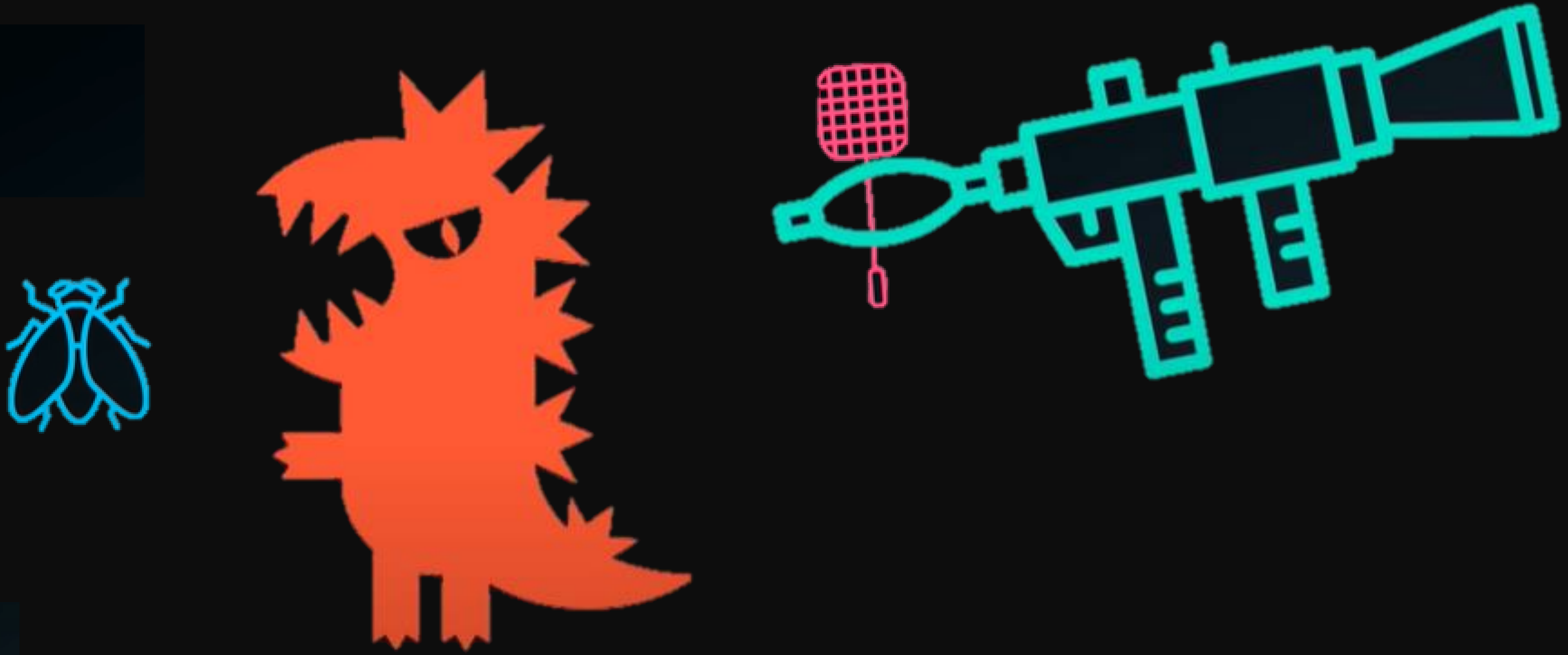
Difference between Overfitting and Underfitting

Difference between Variance and Bais

Detecting Errors

Types of Errors

UNDERFITTING



OVERFITTING

Detecting Errors

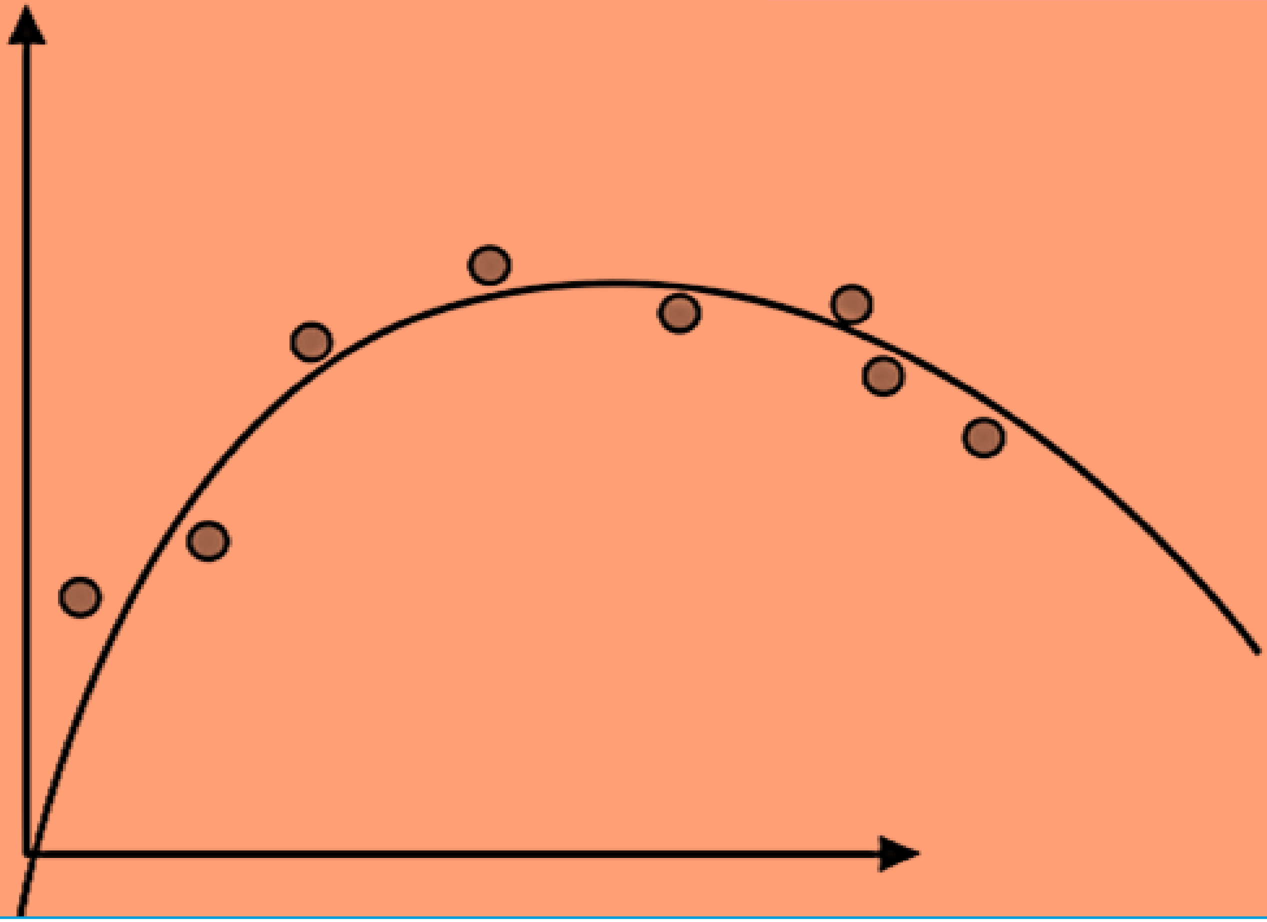
Detecting Errors

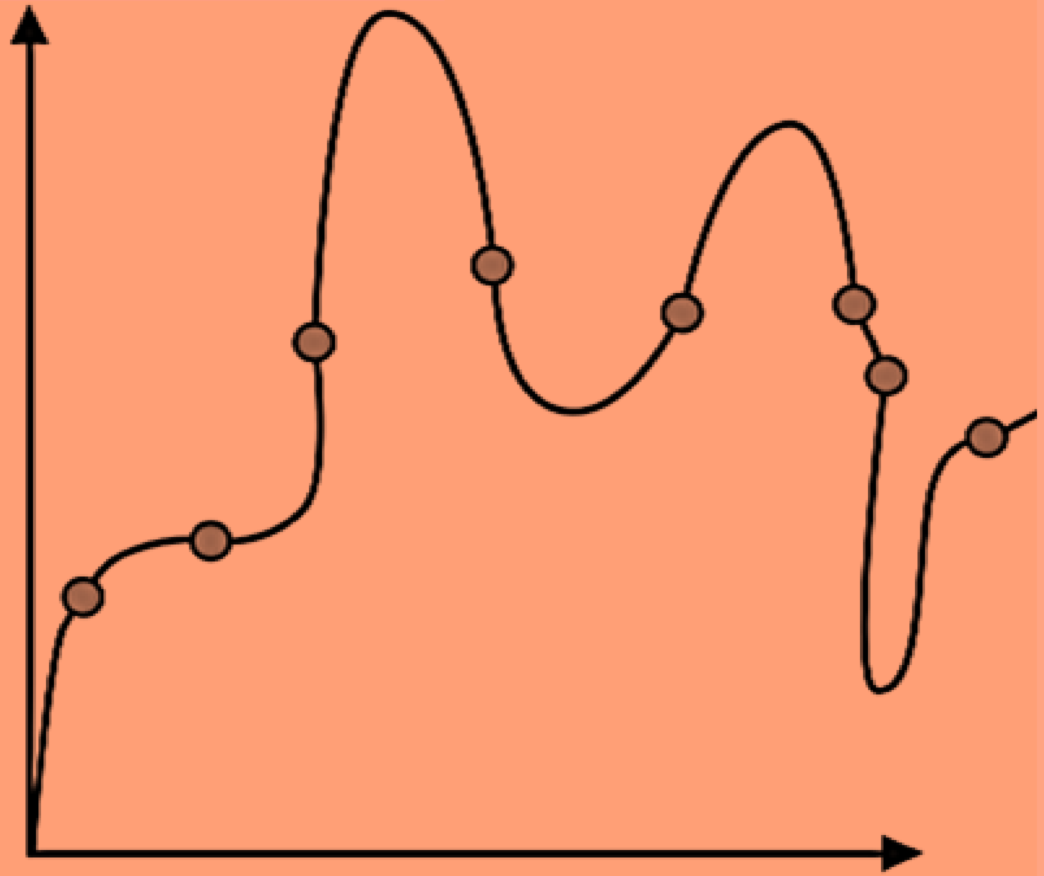
- OVERCOMPLICATED

- OVERFITTING

Error due to variance

This model performs poorly in the testing set

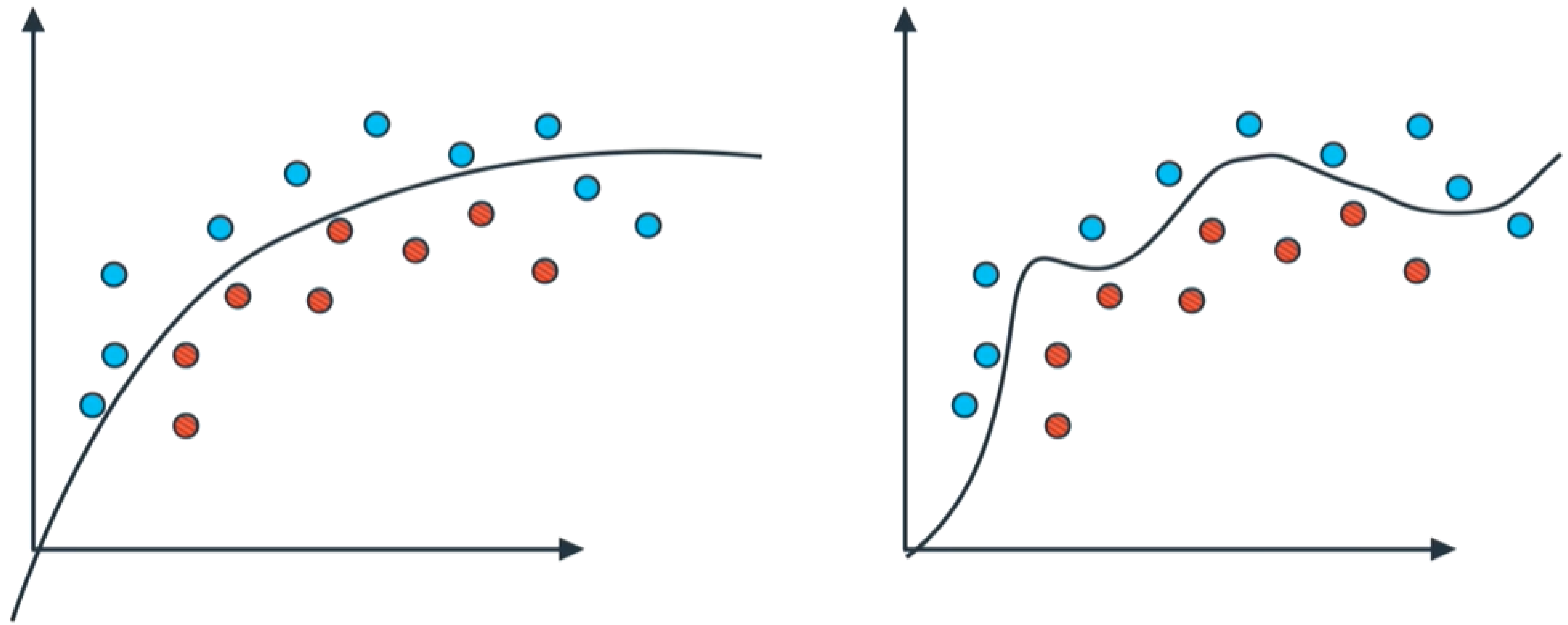




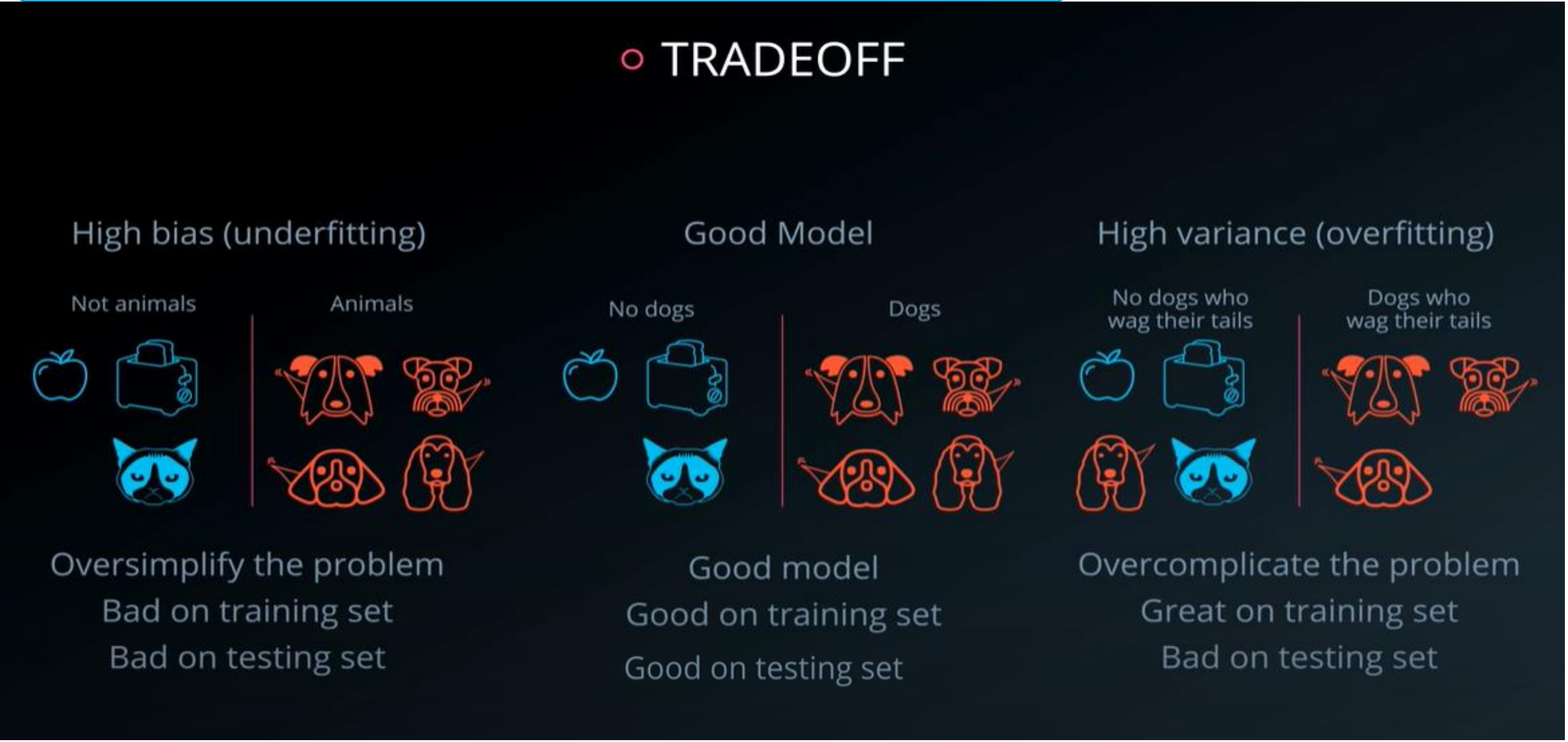
Detecting Errors

○ OVERFITTING

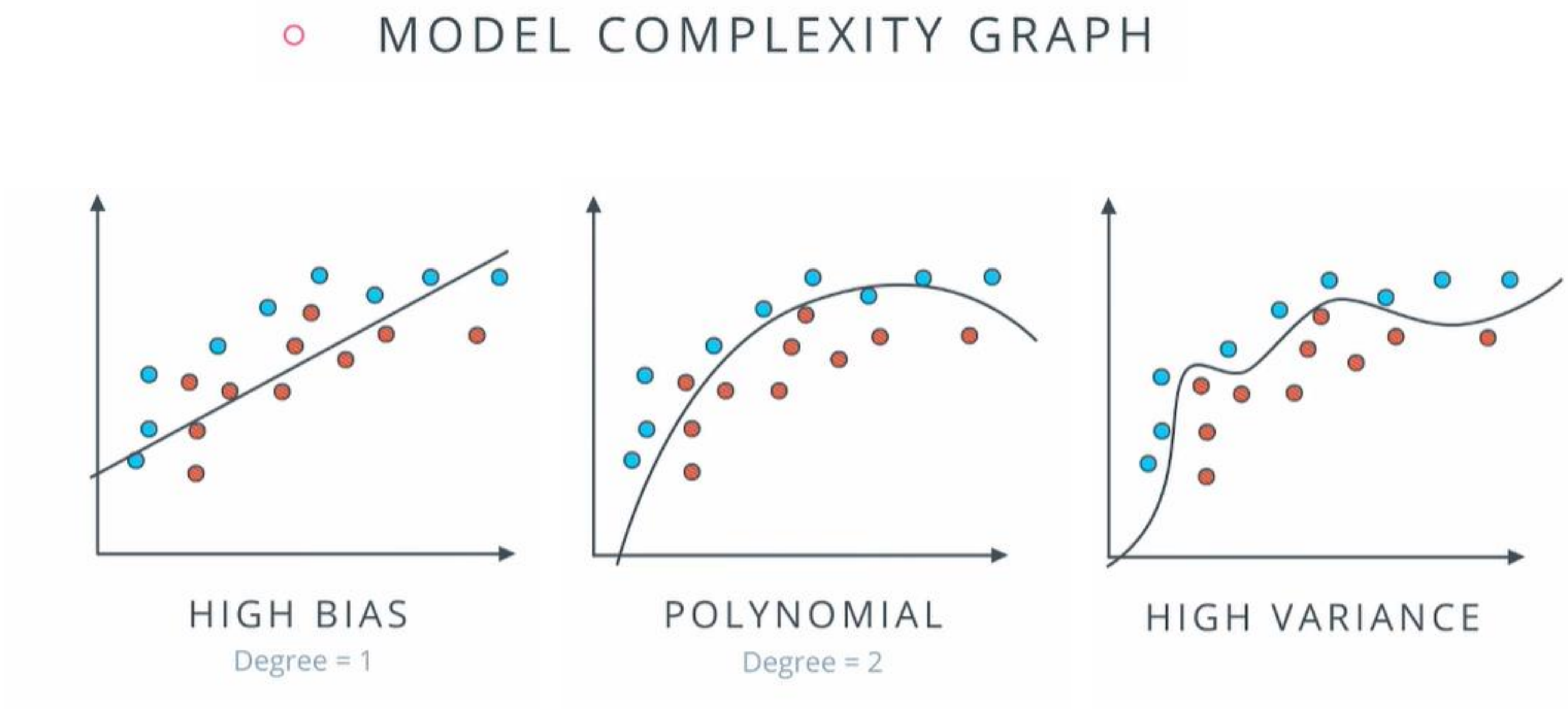
Error due to variance



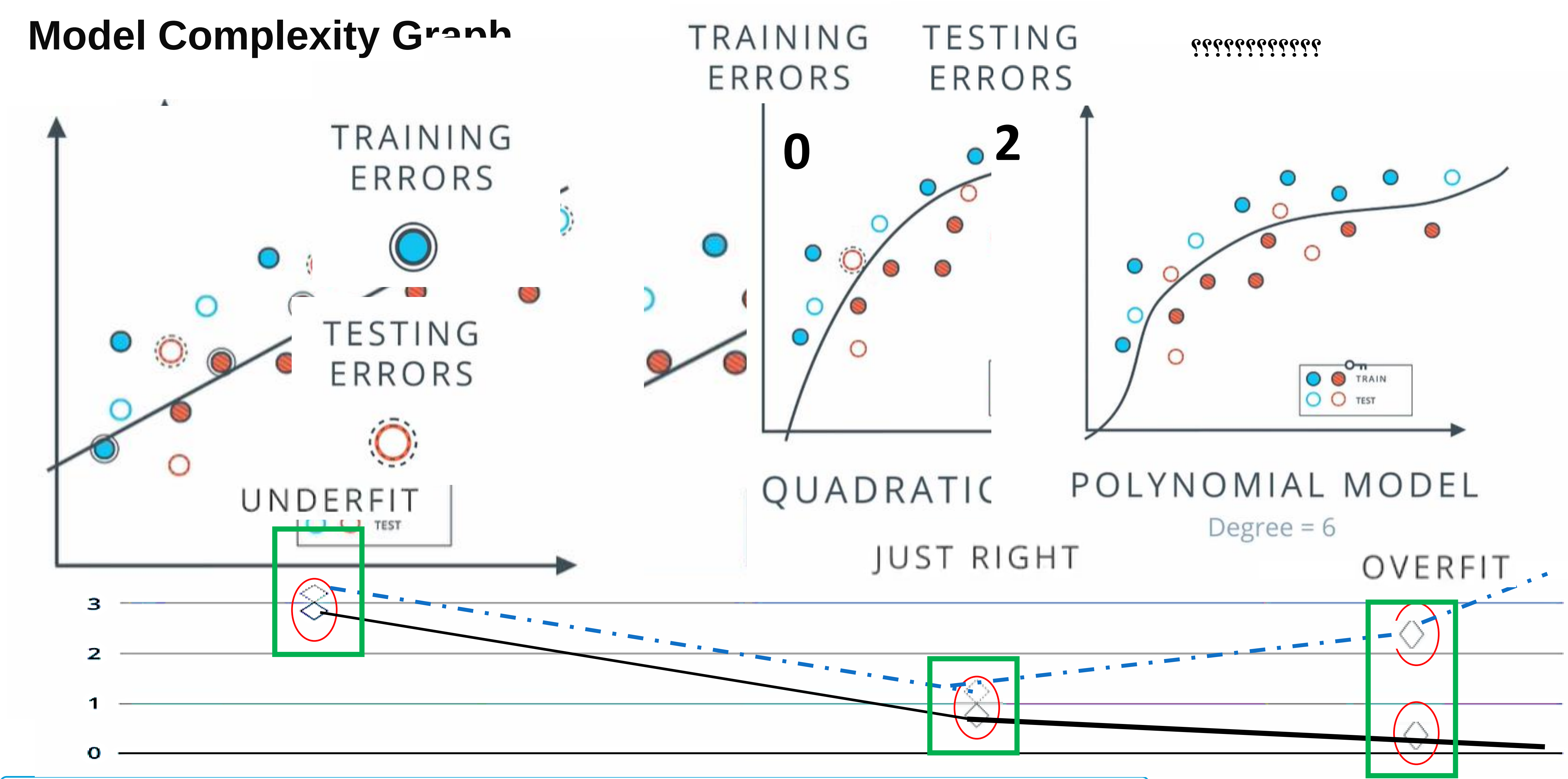
Detecting Errors



Model Complexity Graph



Model Complexity Graph



Supervised Learning-Logistic Regression

- Model will have good accuracy when it is trying to make predictions on new or unseen data

- Good accuracy mean that the value predicted by the model will be very much close to the actual value

- Model will have good accuracy when it is trying to make predictions on new or unseen data

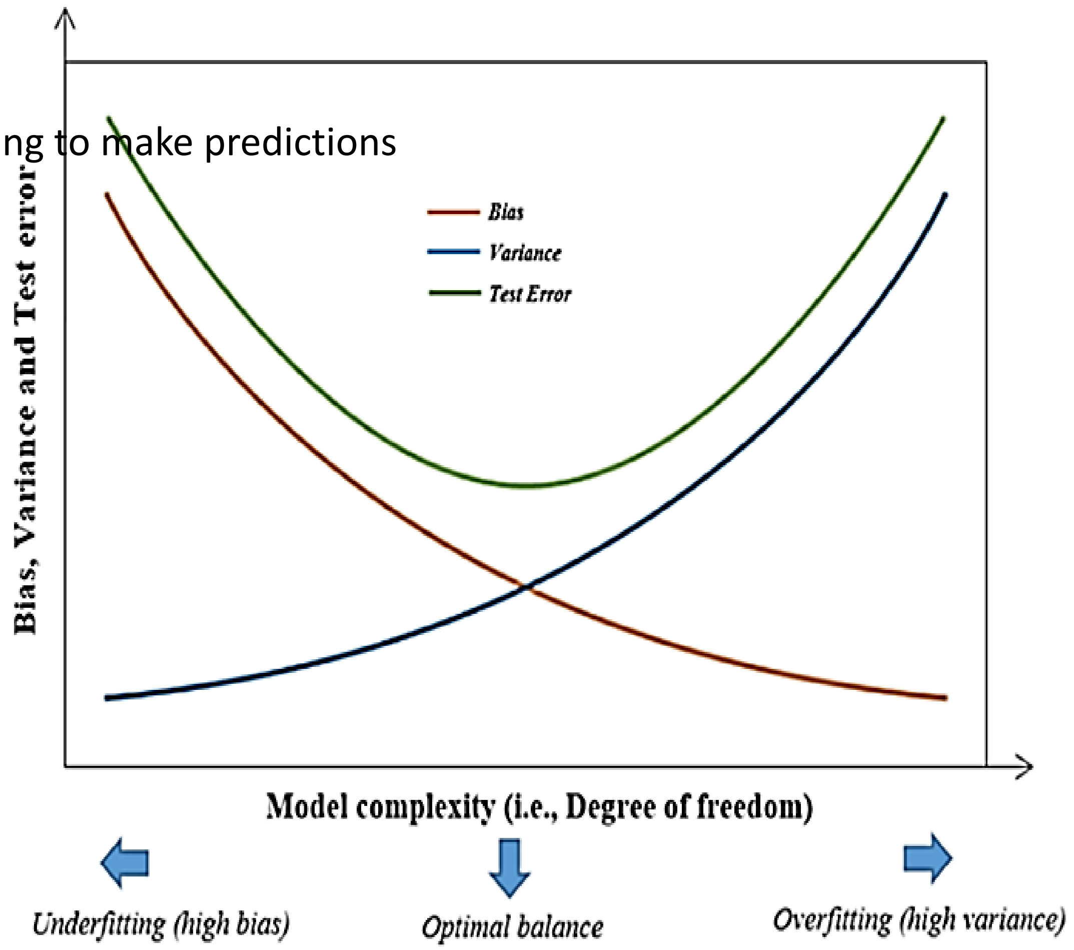
- Bias will be low and variance will be high when model performs well on the training data but perform bad or poor on the test data.

- High variance mean the model can't generalize to new or unseen data(this is the case of Overfitting)

- If the model performs poorly (means less accurate and can not generalize) on both training data and test data. It means it has high bias and high variance (this is the case of Underfitting)

- If model performs well on both test and training data that mean predictions are close to actual values for unseen's data so accuracy will be high so in this case Bias will be low and variance will also be low

- The best model must have low bias (low error rate on training data)and low variance(can generalize and has low error on new or test data)



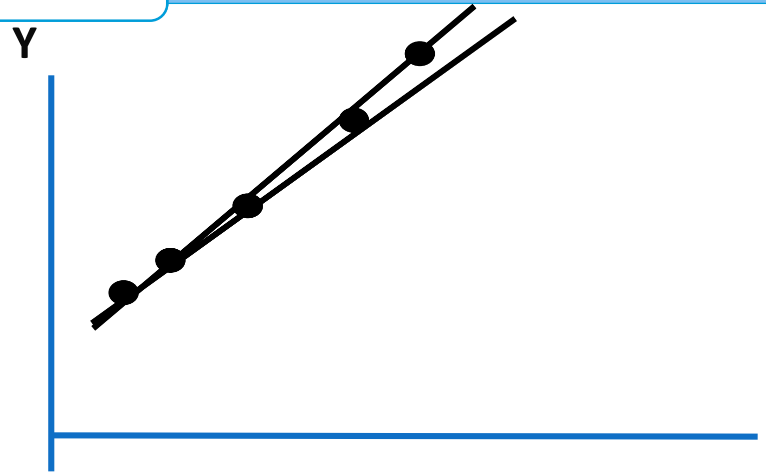
Always have low bias and low variance for your model

Supervised Learning-Regression

Regularization: Ridge and Lasso Regression

Least Absolute Shrinkage and Selection Operator

- a type of linear regression that includes regularization to enhance the model's ability to handle multicollinearity and reduce Overfitting. This technique is particularly useful when dealing with datasets that have a large number of features.



adds a penalty

So for this example how I can change the slope of the line mathematically ???????

Remember we try before to decrease the Overfitting like we used the degree of polynomial

Regularization: Lasso regression adds a penalty equal to the absolute value of the magnitude of coefficients. This penalty term is controlled by a parameter, λ,

Objective Function: The objective function for lasso regression includes both the sum of squared errors and the regularization term:

$$\text{Minimize}\left(\sum_{i=1}^{n}\left(y_i-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda\sum_{j=1}^{p}|\beta_j|\right)$$

$$\sum_{i=1}^{M}(y_i-\hat{y}_i)^2=\sum_{i=1}^{M}\left(y_i-\sum_{j=0}^{p}w_j\times x_{ij}\right)^2+\lambda\sum_{j=0}^{p}w_j^2$$

where y_i is the actual value, x_{ij} is the predictor variable, β_j is the coefficient, and λ is the regularization parameter.

Feature Selection: The regularization term in lasso can shrink some coefficients to exactly zero. This makes lasso not only a tool for regression but also a method for feature selection.

Supervised Learning-Regression

Benefits

•**Reduction of Overfitting**: By adding a penalty for large coefficients, lasso regression can prevent Overfitting, especially in models with many predictors.

•**Feature Selection**: Lasso automatically selects important features, setting less important ones to zero. This simplifies the model and enhances interpretability.

•**Handling Multicollinearity**: By shrinking coefficients, lasso regression can handle multicollinearity among predictors effectively.

Drawbacks

•**Bias**: While reducing variance, lasso can introduce bias into the model, especially when the true underlying model has many predictors with small to moderate effects.

•**Selection Sensitivity**: The selection of features can be sensitive to the value of λ.

Supervised Learning-Logistic Regression

Example for math steps

X1	X2	Y
1	2	4
2	3	5
3	4	6
4	5	7

$y=\beta_0+\beta_1 x_1+\beta_2 x_2$

Ordinary Least Squares (OLS) Solution The goal in OLS is to minimize the sum of squared residuals:

$$\text{Minimize} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{1i} - \beta_2 x_{2i})^2$$

let's assume the OLS solution gives us β1=1 and β2=1.

In lasso regression, we add a penalty for the absolute values of the coefficients. The objective function becomes:

$$\text{Minimize} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{1i} - \beta_2 x_{2i})^2 + \lambda(|\beta_1| + |\beta_2|)$$

Supervised Learning-Logistic Regression

Let's assume λ=1.

Calculating the Lasso Objective

Given our data and the OLS solution, we calculate the lasso objective:

$$\text{Residual Sum of Squares} = \sum_{i=1}^{4}(y_i - \beta_0 - \beta_1 x_{1i} - \beta_2 x_{2i})^2$$

With β0=0, β1=0.5, and β2=0.5:

Residual Sum of Squares=$(6-(1+2))^2+$ $(5-(2+3))^2+$ $(6-(3+4))^2+$ $(7-(4+5))^2$

Residual Sum of Squares=$(4-(3))^2+$ $(5-(5))^2+$ $(6-(7))^2+$ $(7-(9))^2$

$=1^2 + 0^2 + (-1)^2 + (-2)^2$

=1+0+1+4

=6

Now, we add the penalty term $\lambda(|\beta_1| + |\beta_2|)$:

Lasso Objective=6+ 1(|1|+|1|)

=6+2

=8

X1	X2	Y
1	2	4
2	3	5
3	4	6
4	5	7

Supervised Learning-Logistic Regression