Unit III

Unit III REGRESSION AND GENERALIZATION

- Regression: Assessing performance of Regression – Error measures, Overfitting and Underfitting, Catalysts for Overfitting, VC Dimensions
- Linear Models: Least Square method, Univariate Regression, Multivariate Linear Regression, Regularized Regression – Ridge Regression and Lasso
- Theory of Generalization: Bias and Variance Dilemma, Training and Testing Curves Case Study of Polynomial Curve Fitting.

Regression

+ 1   y   1   0
- 1   4   2   1
          3   2

- Regression learning problem is to learn a function estimator.
- Regressor also called as function estimator .
- A mapping

$$\underline{\underline{\hat{f}}} = \boxed{\underline{\underline{\mathfrak{X}}}} \to \boxed{\underline{\underline{\mathbb{R}}}}$$

features

supervised

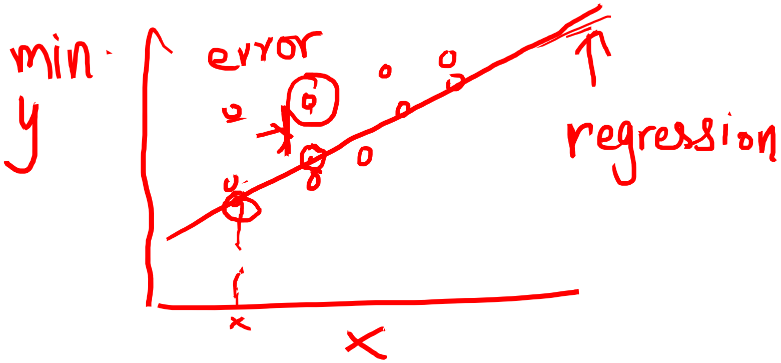
cars

colour | seats | — | price

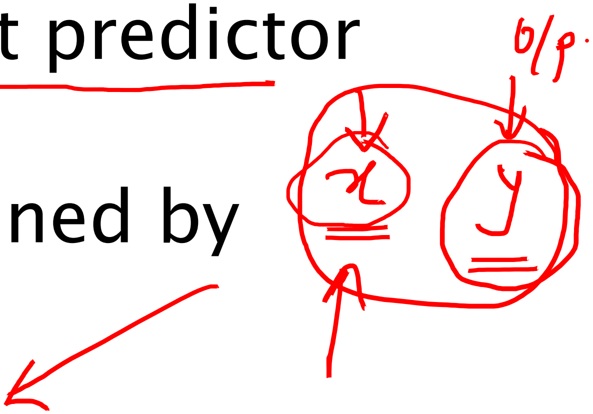
100m

10

Linear Regression



- It is a simple model
- Defines the relationship between a dependent variable and single independent predictor variable.
- Generally these models are defined by equation of line



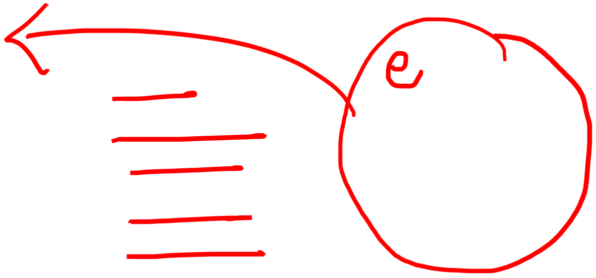
- $\underline{y=\alpha+\beta x}$ or $\underline{y=\overset{c}{\boxed{\beta_0}}\overset{+}{+}\overset{mx}{\boxed{\beta_1}}x}$
- Performing regression analysis involves finding parameters α and β or β_0 and β_1

univariate

- Residuals /error
  - The difference between actual and estimated function values on the training examples

$$E_i = f(x_i) - \hat{f}(x_i)$$

- Least Squares method Lm olm
  - Introduced by Carl Friedrich Gauss
- Finding a line that minimizes the sum of squared residuals.



-

$\bar{y}$ = mean y = $\frac{\sum_{i=1}^{n} y_i}{n}$

$\bar{x}$ = mean x = $\frac{\sum_{i=1}^{n} x_i}{n}$

$$\underline{\beta_0} = \underline{\bar{y}} - \underline{\beta_1 \bar{x}}$$

$$\boxed{\beta_1} = \frac{\sum \left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sum \left(x_i - \bar{x}\right)^2}$$

$$\underline{\hat{y}} = \underline{\beta_0} + \underline{\beta_1} \underline{x}$$

82 → y = ?

$\hat{y} = 26.846 + 0.643 \cdot (X)$

$= 26.846 + 0.643 * 82$

$= 79. - - -$

	Marks in 10th (X)	Marks in 12th (Y)	① $x_i - \bar{x}$	$(x_i - \bar{x})^2$	② $(y_i - \bar{y})$	1*2
1	95	85	17	289	8	136
2	85	95	07	49	18	126
3	80	70	2	4	-7	-14
4	70	65	-8	64	-12	96
5	60	70	-18	324	-7	126
	$\Sigma X = 390$	$\Sigma y = 385$		$\Sigma = 730$		Σ 470

$\bar{x} = 390/5 = 78$ ✓

$\bar{y} = 385/5 = 77$

✓ $$B_1 = \frac{\Sigma(x_i - \bar{x}) * (y_i - \bar{y})}{\Sigma(x_i - \bar{x})^2} = \frac{470}{730}$$

$= 0.643$

✓ $B_0 = \bar{y} - B_1 \cdot \bar{x}$

$= 77 - (0.644) \times 78$

$= 26.846$

$\hat{y} = 26.846 + 0.643 * (X)$

80 ↔ 8 ↔ 78.2

β_0 β_1

Bill Amt = 100

Tip = ?

SSE = ?

Bill Amt	Tip (y)	$\hat{y}$		$(y_i - \hat{y}_i)^2$	
34	5	$0.1462 * 34 - 0.818 = 4.15$			
108	17	108			
64	11	64			
88	8				
99	14				
51	5				

SST = SSE + SSR

sq. sum.

$\bar{y} = 10$

$\bar{x} = 74$

$\beta_1 = 0.1462$

$\beta_0 = -0.818$

$\bar{y} = -0.818 + \beta_1 x$

$= -0.818 + 0.1462 * 100$

$$SSR = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$$

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = 13.8$$

$$r^2 = \frac{SSR}{SST}$$

Least Square Method

The goal of Least Square Method is to find value2 of β$_0$ and β$_1$ for which SSE is minimum

→ Sum of squared Errors

Actual - Predi-

SSE (Sum of Squared Errors)=$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
MSE(Mean Squared Errors)=$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
RMSE (Root Mean Squared Error)=$\sqrt{MSE}$= $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$
R Squared = $1 - \frac{SSE}{VAR(y_i)}$

$$\underline{\underline{SSE}} = \sum_{i=1}^{n}(e_i)^2 = \sum_{i=1}^{n}(y_i - \underline{\underline{\hat{y}_i}})^2 \qquad \hat{y}_i = \beta_0 + \beta_1 x_i$$

$$= \sum_{i=1}^{n}\left(y_i - (\beta_0 + \beta_1 x_i)\right)^2$$

$$= \sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i)^2$$

partial derivatives of SSE w.r.t β_0 & β_1 equal to zero.

$$\frac{\partial SSE}{\partial \beta_0} = 0 \qquad \frac{\partial SSE}{\partial \beta_1} = 0$$

$$\frac{\partial SSE}{\partial \beta_0} = 0$$

$$\frac{\partial\left(\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i)^2\right.}{\partial \beta_0} = 0$$

$$\sum_{i=1}^{n}\frac{\partial}{\partial \beta_0}(y_i - \beta_0 - \beta_1 x_i)^2 = 0$$

$$\sum_{i=1}^{n} -2(y_i - \beta_0 - \beta_1 x_i) = 0$$

$$-2\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i) = 0$$

$$\sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_1) = 0 \quad \text{———} \quad \textcircled{1}$$

$$\frac{\partial SSE}{\partial \beta_1} = \frac{\partial}{\partial \beta_1} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2 = 0$$

$$\sum_{i=1}^{n} \frac{\partial}{\partial \beta_1} (y_i - \beta_0 - \underline{\underline{\beta_1}} x_i)^2 = 0$$

$$\sum_{i=1}^{n} -2x_i (y_i - \beta_0 - \beta_1 x_i) = 0$$

$$* \quad \sum_{i=1}^{n} x_i (y_i - \beta_0 - \beta_1 x_i) = 0 \quad \text{—} \quad \textcircled{2}$$

$$\sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i) = 0 \quad \text{—} \quad \textcircled{1}$$

$$\sum_{i=1}^{n} y_i - \sum_{i=1}^{n} \beta_0 - \beta_1 \sum_{i=1}^{n} x_i = 0$$

$$\sum_{i=1}^{n} y_i - \beta_0 \sum_{i=1}^{n} 1 - \beta_1 \sum_{i=1}^{n} x_i = 0$$

$$\sum_{i=1}^{n} y_i - \beta_0 n - \beta_1 \sum_{i=1}^{n} x_i = 0$$

$$n \cdot \beta_0 = \sum_{i=1}^{n} y_i - \beta_1 \sum_{i=1}^{n} x_i$$

$$\beta_0 = \underbrace{\frac{1}{n} \cdot \sum_{i=1}^{n} y_i}_{\bar{Y}} - \underbrace{\frac{\beta_1}{n} \sum_{i=1}^{n} x_i}_{\bar{X}}$$

$$\boxed{\beta_0 = \bar{Y} - \beta_1 \bar{X}} \longleftarrow$$

eq^n ②

$$\sum_{i=1}^{n} x_i \left(y_i - \underline{\beta_0} - \beta_1 x_i \right) = 0$$

$$\sum_{i=1}^{n} x_i \left(y_i - \bar{Y} + \beta_1 \bar{X} - \beta_1 x_i \right) = 0$$

$$\sum_{i=1}^{n} x_i (y_i - \bar{Y} + \underleftarrow{\beta_1} (\bar{x} - x_i) = 0$$

$$\sum_{i=1}^{n} x_i\left(y_i - \bar{Y} - \beta_1\left(x_i - \bar{X}\right)\right) = 0$$

$$\sum_{i=1}^{n} x_i\left(y_i - \bar{Y}\right) - \sum_{i=1}^{n} x_i \beta_1 \left(x_i - \bar{X}\right) = 0$$

$$\sum_{i=1}^{n} x_i\left(y_i - \bar{Y}\right) = \beta_1 \sum_{i=1}^{n} x_i\left(x_i - \bar{x}\right)$$

$$\beta_1 = \frac{\sum_{i=1}^{n} x_i\left(y_i - \bar{Y}\right)}{\sum_{i=1}^{n} x_i\left(x_i - \bar{X}\right)} = \frac{\sum_{i=1}^{n} \left(x_i - \bar{X}\right)\left(y_i - \bar{Y}\right)}{\sum_{i=1}^{n} \left(x_i - \bar{X}\right)\left(x_i - \bar{X}\right)}$$

$$\boxed{\beta_1 = \frac{\sum_{i=1}^{n} \left(x_i - \bar{x}\right)\left(Y_i - \bar{Y}\right)}{\sum_{i=1}^{n} \left(x_i - \bar{x}\right)^2}}$$

SSE, MSE, r squared, RMSE

SST = sum of squares of Total $= \sum_{i=1}^{n} (y_i - \bar{Y})^2$

SSR = sum of squares due to regression.

SSR = SST - SSE

- The estimation of unknown parameters using appropriate method provides the values of the parameter. Substituting these values in the equation gives us a usable model. This is termed as model fitting.

Steps in regression analysis

- Statement of the problem under consideration
- Choice of relevant variables
- Collection of data on relevant variables
- Specification of model ← car price
- Choice of method for fitting the data
- Fitting of model
  - Least square method, maximum likelihood method, ridge method, principal components method
- Model validation and criticism ←
- Using the chosen model(s) for the solution of the posed problem

Multivariate Linear Regression

- **Multivariate Linear Regression**
  Similar to the simple linear regression model but with multiple independent variables contributing to the dependent variable
- hence multiple coefficients to determine and complex computation due to the added variables.

$$\hat{y} = \beta_0 + \boxed{\underline{\beta_1}}\, \underline{x_1} + \boxed{\beta_2}\, x_2 +$$

$$\hat{y} \longleftarrow x_1 \quad x_2 \quad x_3 \quad \cdots \quad x_n$$

Multivariate Linear Regression

1) 2) 3) 4

- The equation of multivariate linear regression,

$$Y_i = \alpha + \text{ß}_1 x_i^{(1)} + \text{ß}_2 x_i^{(2)} + \dots + \text{ß}_n x_n^{(n)}$$

$\beta_0 \qquad x_4^{0} = 1 \qquad x_4^{2} \qquad x_4^{n}$

- Y_i is the estimate of ith component of dependent variable y,
- where we have **n** independent variables and $x_i^{(j)}$denotes the ith component of the jth independent variable/feature.

Multivariate Linear Regression

- Univariate linear regression can be written in matrix format as

$$\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} a + \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} b + \begin{pmatrix} \epsilon_1 \\ \vdots \\ \underline{\underline{\epsilon_n}} \end{pmatrix}$$

$$\underline{\underline{\mathbf{y}}} = \mathbf{a} + \mathbf{X}\mathbf{b} + \underline{\underline{\boldsymbol{\epsilon}}}$$

Multivariate Linear Regression

- By using the same concept as that of univariate regression we have formula for Multivariate Linear Regression

$$Y_i = \underline{\beta_0} + \underline{\beta_1} X_{i,1} + \beta_2 X_{i,2} + \beta_3 X_{i,1} X_{i,2} + \epsilon_i$$

- This can be written as
- $Y = \beta X + \epsilon$

$Y = \beta X + \epsilon$

- Where $Y = \begin{matrix} y1 \\ y2 \\ . \\ . \\ . \\ yn \end{matrix}$

Multivariate Linear Regression

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}$$

- This can be written as
- Y = βX + ϵ
-
- Where Y $= \begin{bmatrix} y1 \\ y2 \\ . \\ . \\ . \\ yn \end{bmatrix}$ X $= \begin{bmatrix} 1 & x11 \ldots.. & x1d \\ 1 & x21 \ldots.. & x2d \\ . & \ldots\ldots\ldots\ldots & x3d \\ . & \ldots\ldots\ldots. & .. \\ . & \ldots\ldots & \ldots \\ 1 & xn1 \ldots\ldots. & xnd \end{bmatrix}$
- B $= \begin{bmatrix} \beta1 \\ \beta2 \\ . \\ . \\ . \\ \beta n \end{bmatrix}$

$$\epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ . \\ \vdots \\ \epsilon_n \end{bmatrix}$$

Errors

- Training Error
- Test Error

Bias and Variance

- Variance is the amount that the estimate of the target function will change if different training data was used
- **Low Variance**: Suggests small changes to the estimate of the target function with changes to the training dataset.
- **High Variance**: Suggests large changes to the estimate of the target function with changes to the training dataset.

- Variance : Average of squared differences between each value and the mean value.

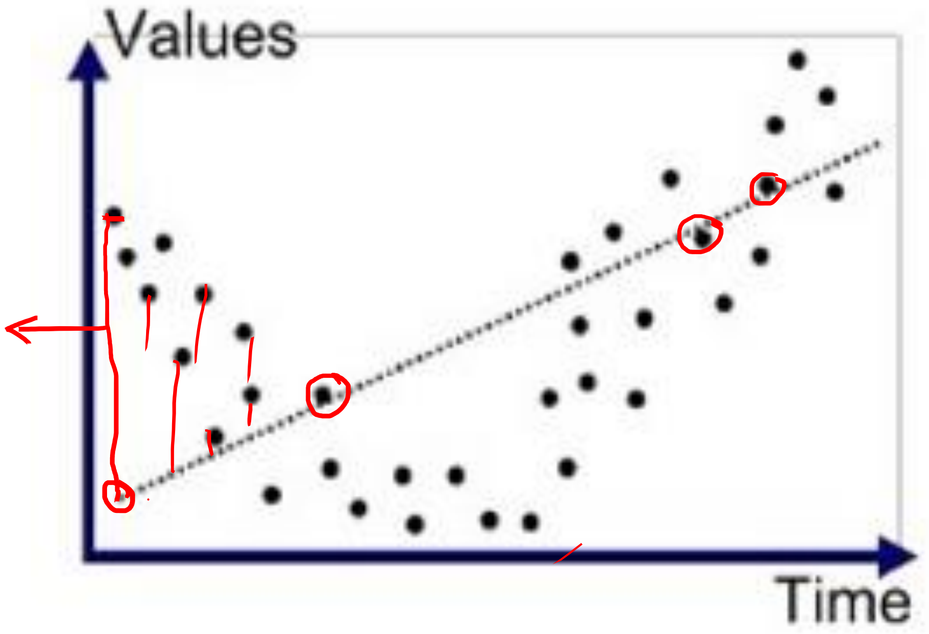
$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2$$

- Standard deviation is the square root of variance

- Bias are the simplifying assumptions made by a model to make the target function easier to learn.
- **Low Bias**: Suggests less assumptions about the form of the target function.
- **High-Bias**: Suggests more assumptions about the form of the target function.

Underfitting

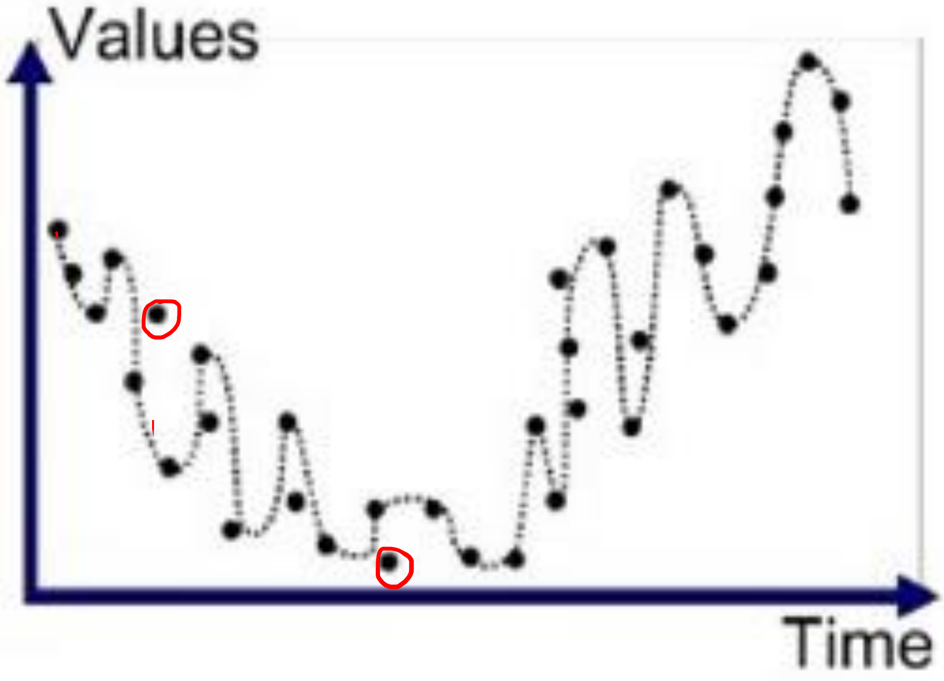
- Predicted values are wrong
- High Bias



Underfitted

Overfitting

- Attempts to fit almost all values
- Complex Model with high degree
- High Variance
- Low Training error
- High Testing Error



Overfitted

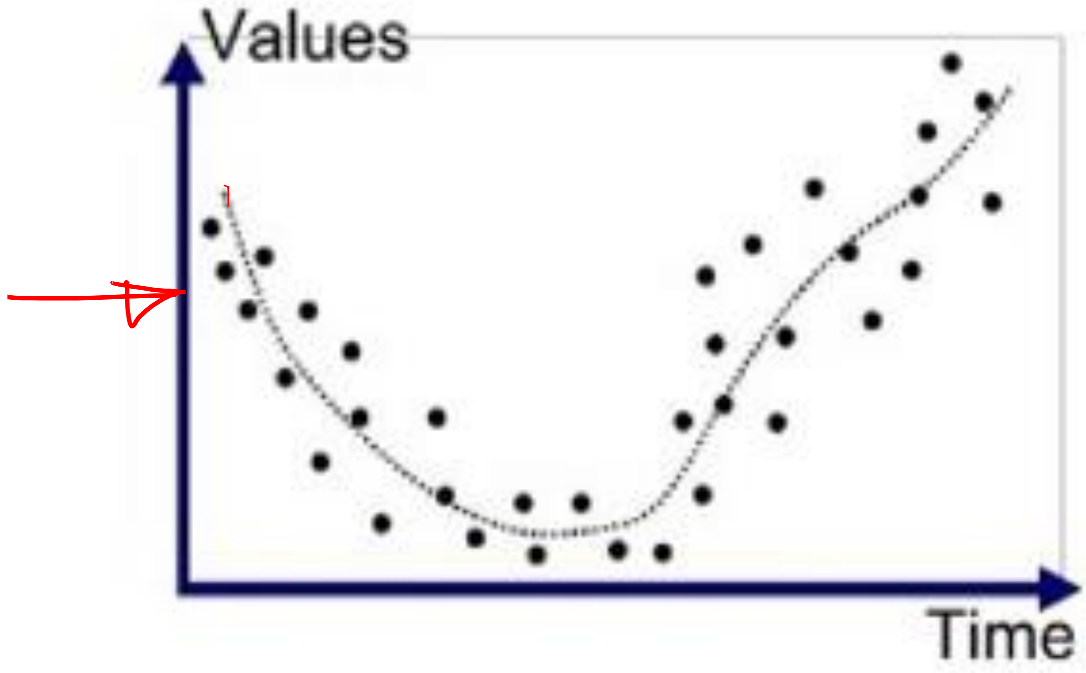
Catalyst of Overfitting

- Noise
- ✓ High Complexity
- ✓ Small Training Set — missing
- ✓ High Number of Features

~~dro~~

drop

Good fit or Just Right Fit



Good Fit/Robust

Bias-Variance Trade-Off

high

bias

- The goal of any supervised machine learning algorithm is to achieve low bias and low variance. In turn the algorithm should achieve good prediction performance.

- Parametric or linear machine learning algorithms often have a high bias but a low variance.
- Non-parametric or non-linear machine learning algorithms often have a low bias but a high variance.
- bias and variance provide the tools to understand the behavior of machine learning algorithms in the pursuit of predictive performance.

Overfitting and underfitting

- An estimated model is said to underfit if it exhibits a large error in prediction. We should try to minimize this error in the model.
- However a formulated model with low error could also indicate that the model doesn't understand the underlying relationship between the features of the model. This could result that the model is memorizing the supplied data. The model is said to overfit.

low

- ✓ An underfit model is said to exhibit high bias and ✓ an overfit model said to exhibit high variance.

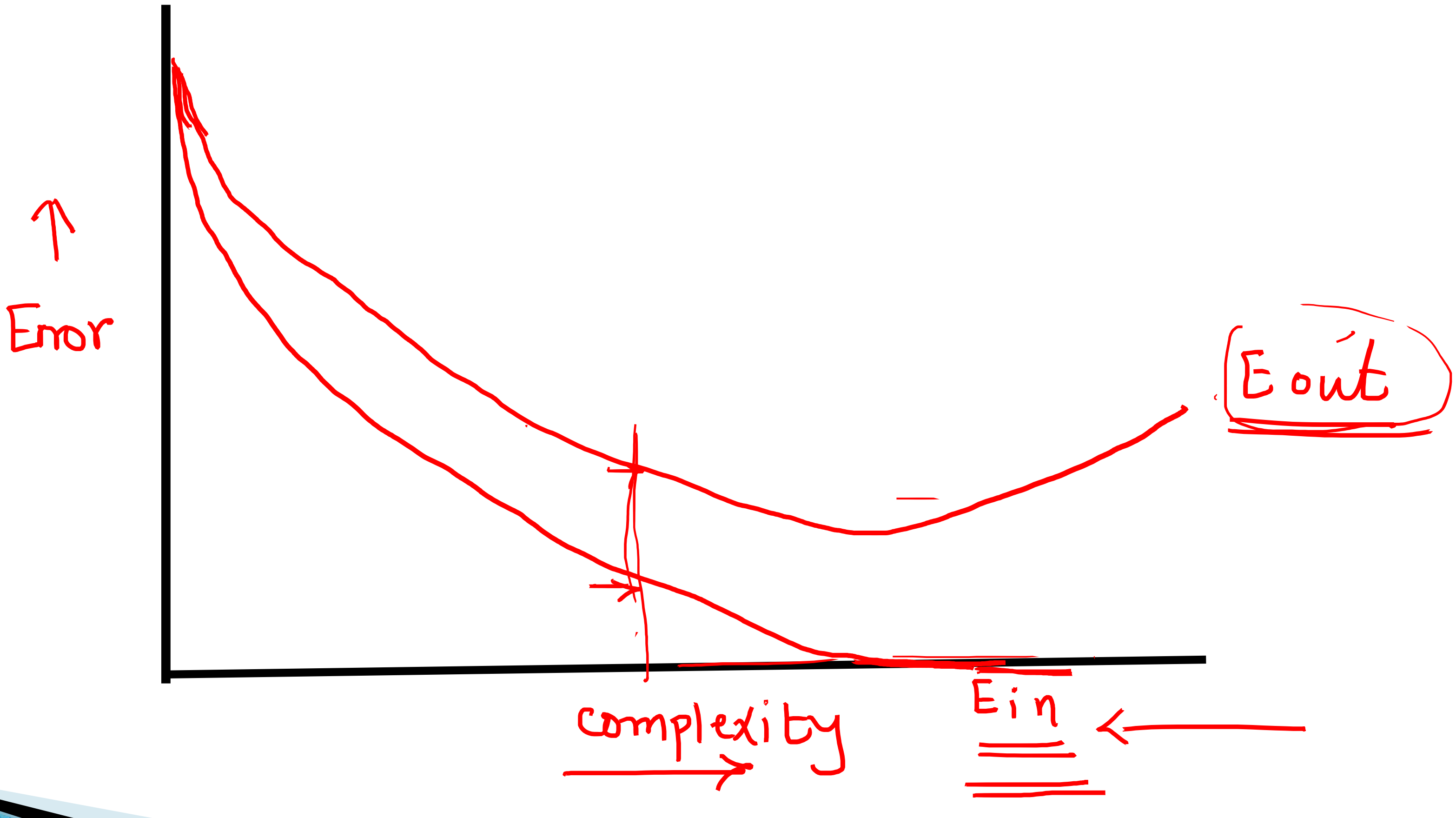
low

good

Ein ←

Eout ←

early stopping



Regularized Regression

ovid overfitting

→ high variance

↓

bias

- Regularization : a method to avoid over fitting by applying additional constraint on weight vector. shrinkage method.
- Least square regression problem can be written as an optimization problem

Multi colinearity

(y - ŷ)

$$\boxed{\beta} = \arg\min(\underline{y - \beta X})^T(\underline{y - X\beta})$$

square of re

Ridge Regression

$Y = X\beta \rightarrow \begin{bmatrix}\beta_1\\ \beta_2\\ \vdots\end{bmatrix}$

- Set β as close as possible to 0
- Reduction of weights results in reduced variance and increase in bias
- Reduces errors in prediction
- Increases the stability of Model

$\cancel{\beta^{rigd}}$ β^{ridge}

$$w^* = \underset{w}{\arg\min}\,(y - Xw)^T(y - Xw) + \boxed{\lambda\|w\|^2}$$

$(-X$

- Using regularized regression

$$w^* = \underset{w}{\arg\min}\,(y - Xw)^T(y - Xw) + \lambda\|w\|^{\boxed{2}}$$

- Which can be further simplified to

$$\underline{\hat{w} = (X^TX + \lambda I)^{-1}X^Ty}$$

where I denotes the identity matrix

- This form of least-squares regression is known as ***ridge regression.***
- This is L2 regularization i.e.. Add the penalty equivalent to square of magnitude

Lasso

- Least Absolute Shrinkage and Selection of Operator
- Sets some values of β to zero
- Performs L1 regularization i.e. add the penalty equal to absolute value of magnitude of coefficient

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} (\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) + \boxed{\lambda||\mathbf{w}||^2}$$

- In Ridge
- COST(w) = RSS(w) + ʎ * (sum of squares of weight)
- In Lasso
- COST(w) = RSS(w) + ʎ * (sum of absolute value of weight)

penalty

Key Differences

Multi colinearity

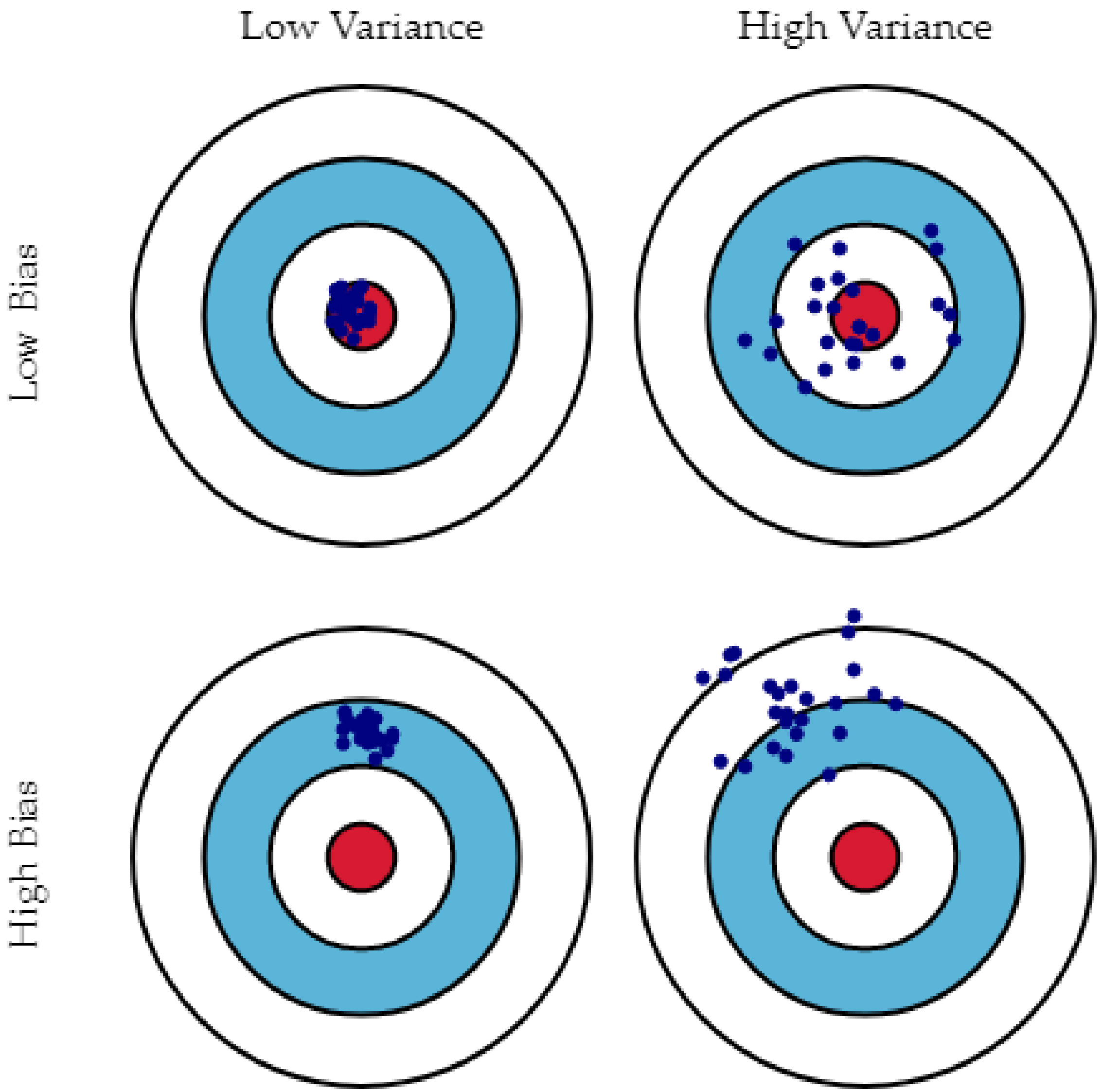
- Ridge
  - L2 Norm
  - Majorly used to prevent overfittig
  - Not useful in case of high no of features
  - Works well when features are highly correlated
- LASSO
  - L1 norm
  - Generally used for where no of features are high
  - Also performs feature selection
  - Does not work well when features are highly correlated

Generalization

- Generalization usually refers to a ML model's ability to perform well on new unseen data rather than just the data that it was trained on.

- **What is bias?**
- Bias is the difference between the average prediction of our model and the correct value which we are trying to predict.
- Model with high bias pays very little attention to the training data and oversimplifies the model.
- It always leads to high error on training and test data.

- **What is variance?**
- Variance is the variability of model prediction for a given data point or a value which tells us spread of our data.
- Model with high variance pays a lot of attention to training data and does not generalize on the data which it hasn’t seen before.
- As a result, such models perform very well on training data but has high error rates on test data.


Low Variance
High Variance
Low Bias
High Bias

- **Mathematically**
- Let the variable we are trying to predict as Y and other covariates as X. We assume there is a relationship between the two such that
- $Y=f(X) + e$
- Where e is the error term and it’s normally distributed with a mean of 0.
- We will make a model f^(X) of f(X) using linear regression or any other modeling technique.

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

The Err(x) can be further decomposed as

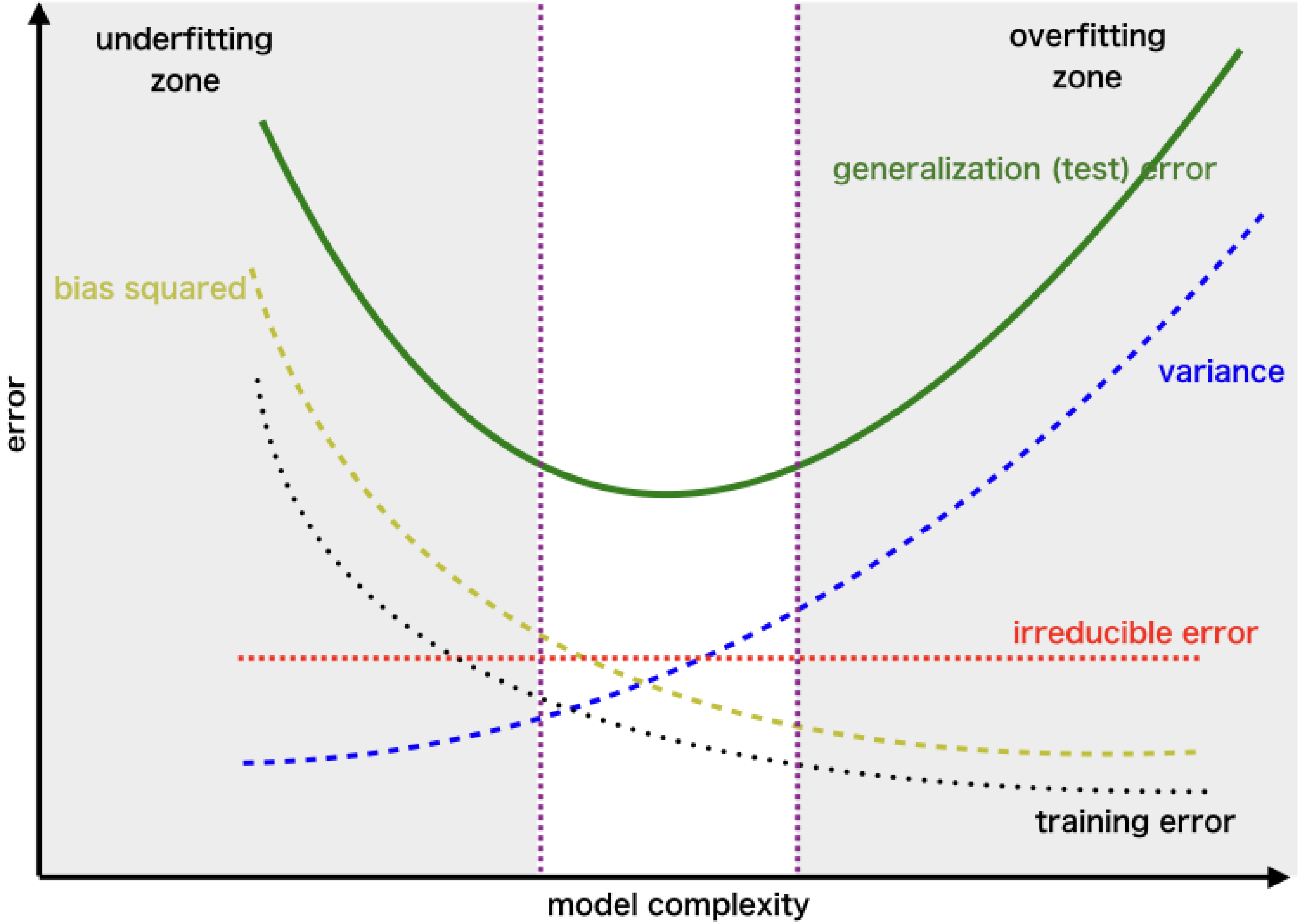
$$Err(x) = \left(E[\hat{f}\,(x)] - f(x)\right)^2 + E\left[\left(\hat{f}\,(x) - E[\hat{f}\,(x)]\right)^2\right] + \sigma_e^2$$

$$Err(x) = \mathrm{Bias}^2 + \mathrm{Variance} + \mathrm{Irreducible\ Error}$$

- In supervised learning, **overfitting** happens when our model captures the noise along with the underlying pattern in data. It happens when we train our model a lot over noisy dataset.
- These models have low bias and high variance.
- These models are very complex and are prone to overfitting.

- In supervised learning, **underfitting** happens when a model unable to capture the underlying pattern of the data.
- These models usually have high bias and low variance.
- It happens when we have very less amount of data to build an accurate model or when we try to build a linear model with a nonlinear data.
- Also, these kind of models are very simple to capture the complex patterns in data like Linear and logistic regression.

- If our model is too simple and has very few parameters then it may have high bias and low variance.
- On the other hand if our model has large number of parameters then it’s going to have high variance and low bias.
- So we need to find the right/good balance without overfitting and underfitting the data.
- This tradeoff in complexity is why there is a tradeoff between bias and variance.
- An algorithm can’t be more complex and less complex at the same time.
- To build a good model, we need to find a good balance between bias and variance such that it minimizes the total error.


underfitting zone
overfitting zone
generalization (test) error
bias squared
variance
irreducible error
training error
error
model complexity

Generalization

- If you over train the model on the training data, then it will be able to identify all the relevant information in the training data, but will fail miserably when presented with the new data.
- We then say that the model is incapable of *generalizing*, or that it is *overfitting* the training data.

- To create good predictive models in machine learning that are capable of generalizing, one needs to know when to stop training the model so that it doesn't overfit.

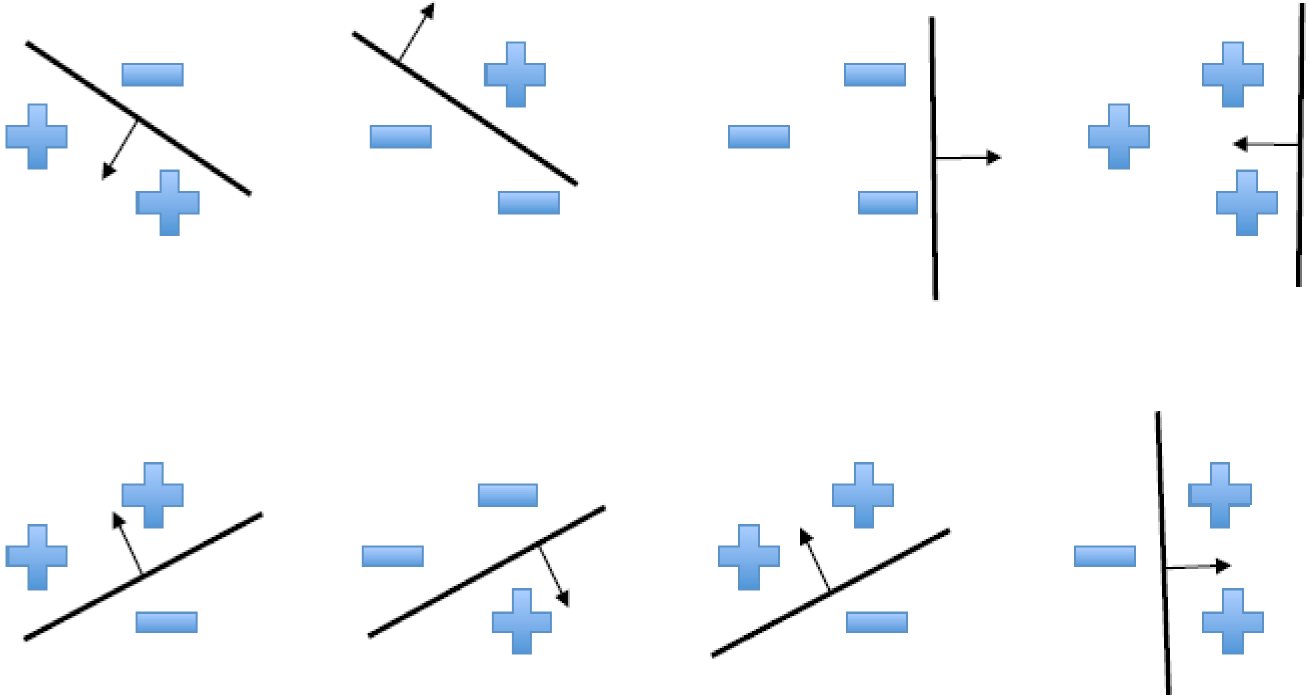
VC dimension

- A dataset containing N points.
- N points can be labeled in 2^N ways as positive and negative.
- 2^N different learning problems can be defined by N data points.
- If for any of these problems, we can find a hypothesis $h \in H$ that separates the positive examples from the negative, then we say H shatters N points.
- That is, any learning problem definable by N examples can be learned with no error by a hypothesis drawn from H.

- The maximum number of points that can be shattered by H is called the Vapnik–Chervonenkis (VC) dimension of H, is denoted as VC(H), and measures the capacity of H.

An Example of VC Dimension

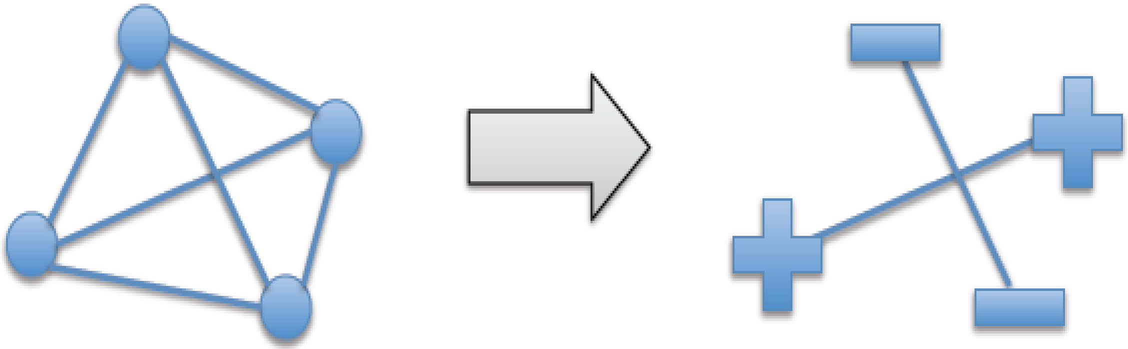
- Suppose our model class is a hyperplane
- Consider all labelings over three points in $\mathbb{R}^2$



- In $\mathbb{R}^2$, we can find a plane (i.e., a line) to capture any labeling of 3 points. A 2D hyperplane shatters 3 points

An Example of VC Dimension

- But, a 2D hyperplane cannot deal with some labelings of four points:



Connect all pairs of points; two lines will always cross

Can’t separate points if the pairs that cross are the same class

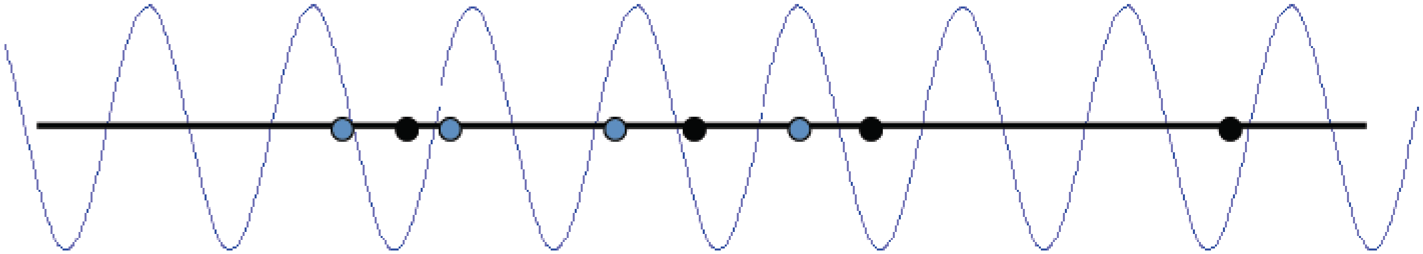
- Therefore, a 2D hyperplane cannot shatter 4 points

Some Examples of VC Dimension

- The VC dimension of a hyperplane in 2D is 3.
  - In d dimensions it is d+1
    - It's just a coincidence that the VC dimension of a hyperplane is almost identical to the # parameters needed to define a hyperplane

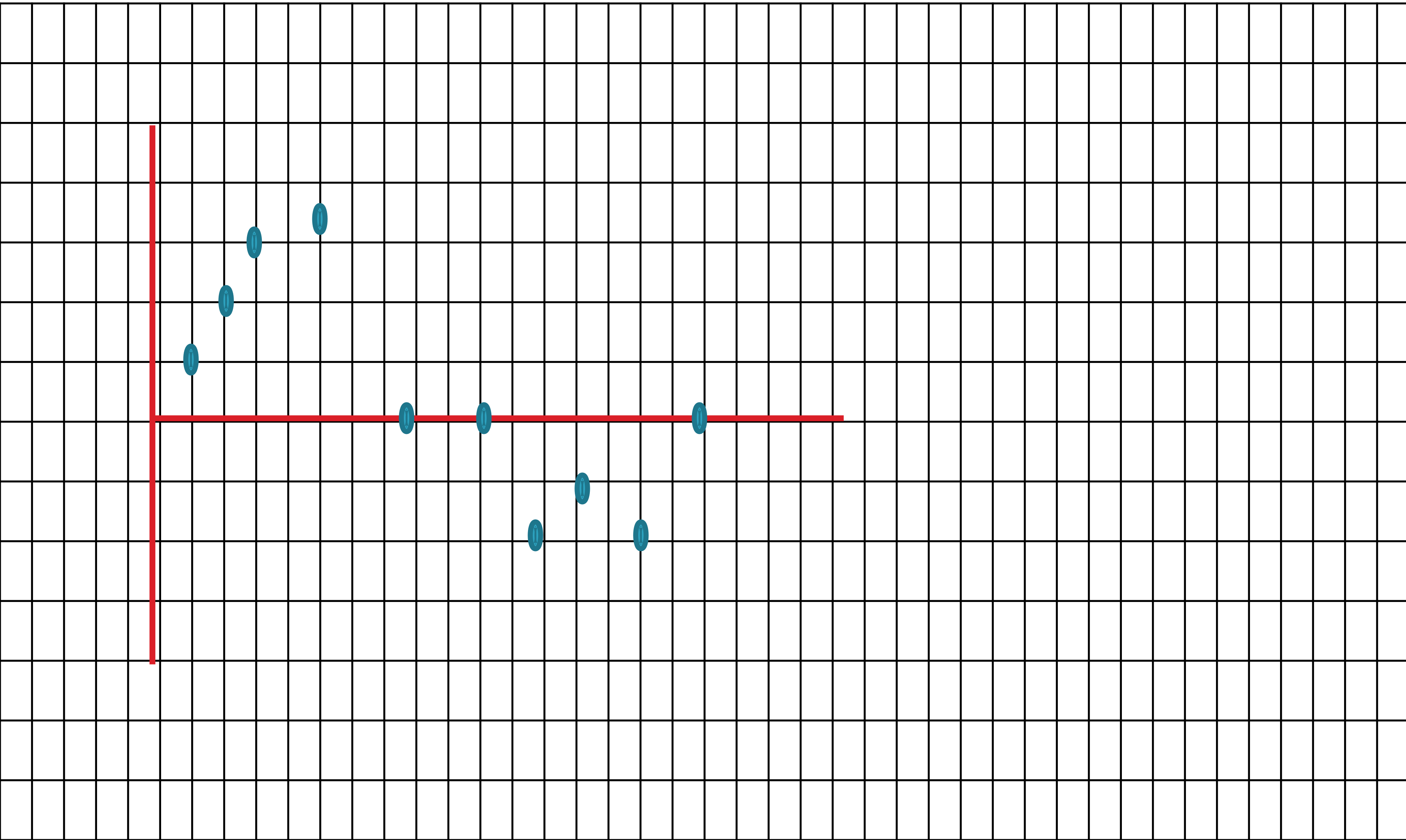
- A sine wave has infinite VC dimension and only 2 parameters!
  - By choosing the phase & period carefully we can shatter any random set of 1D data points (except for nasty special cases)

$$h(x) = a\sin(bx)$$

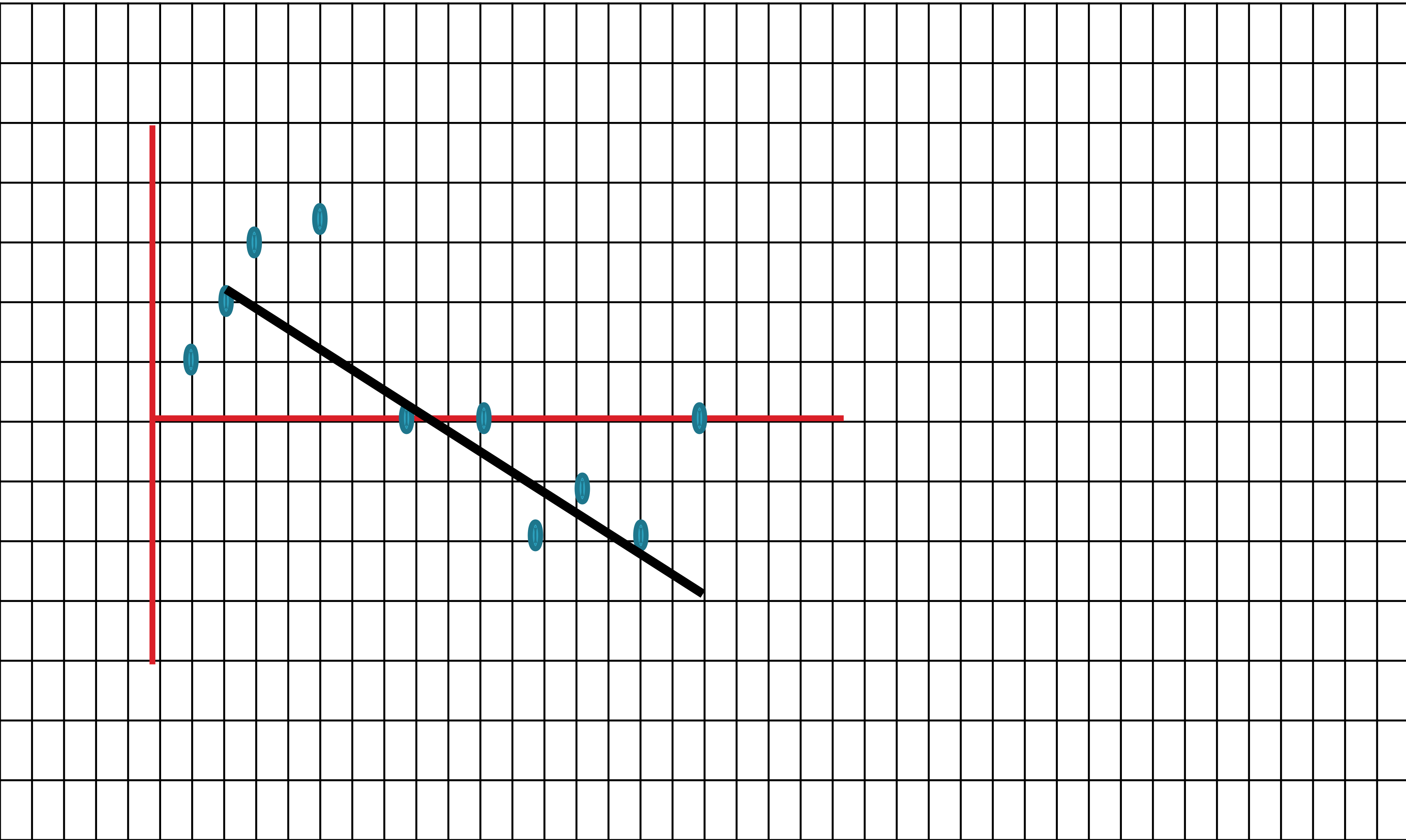


- VC dimension(d_{vc}) sets the thumb rule
- $N \geq 10 . d_{vc}$

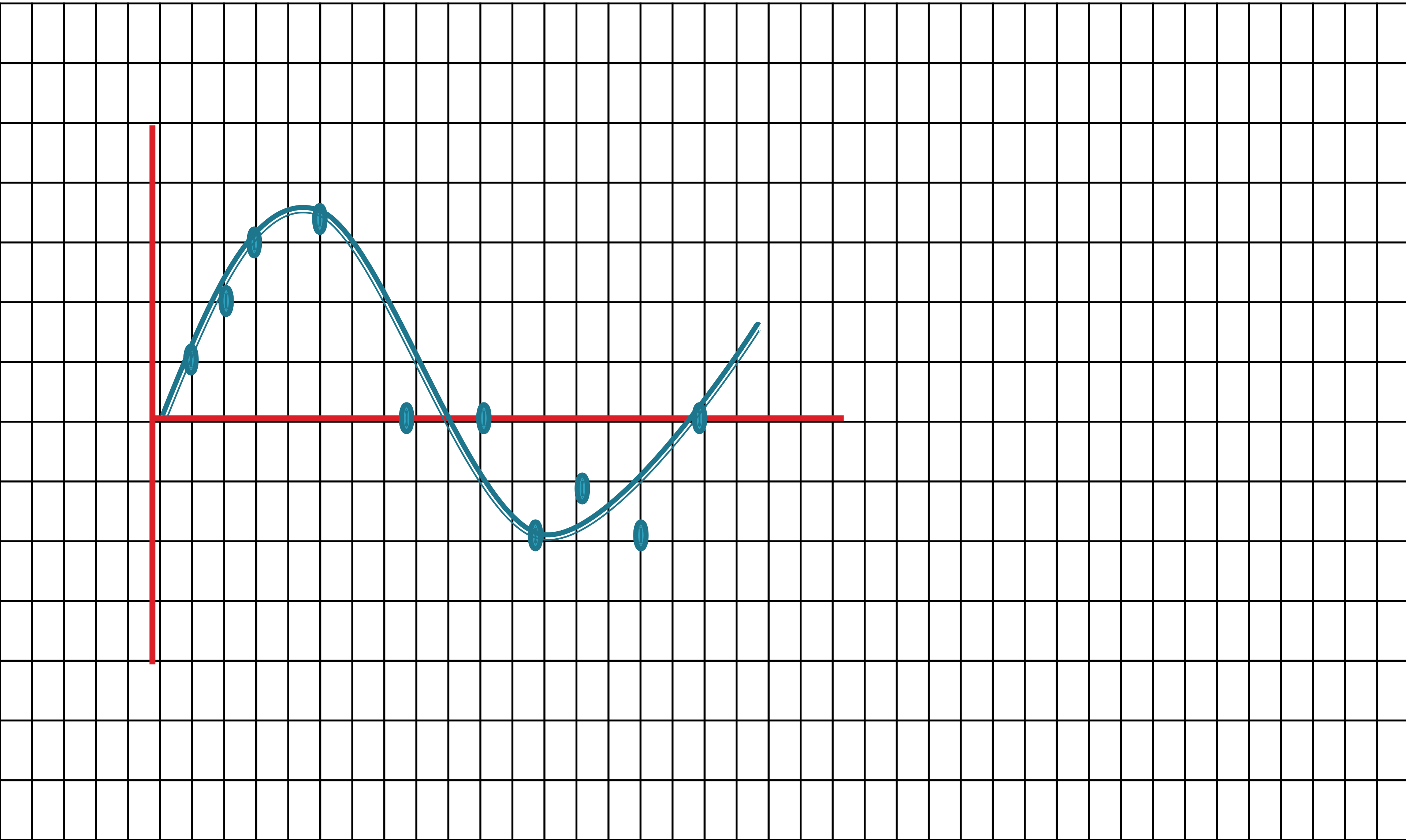
Polynomial Curve Fitting



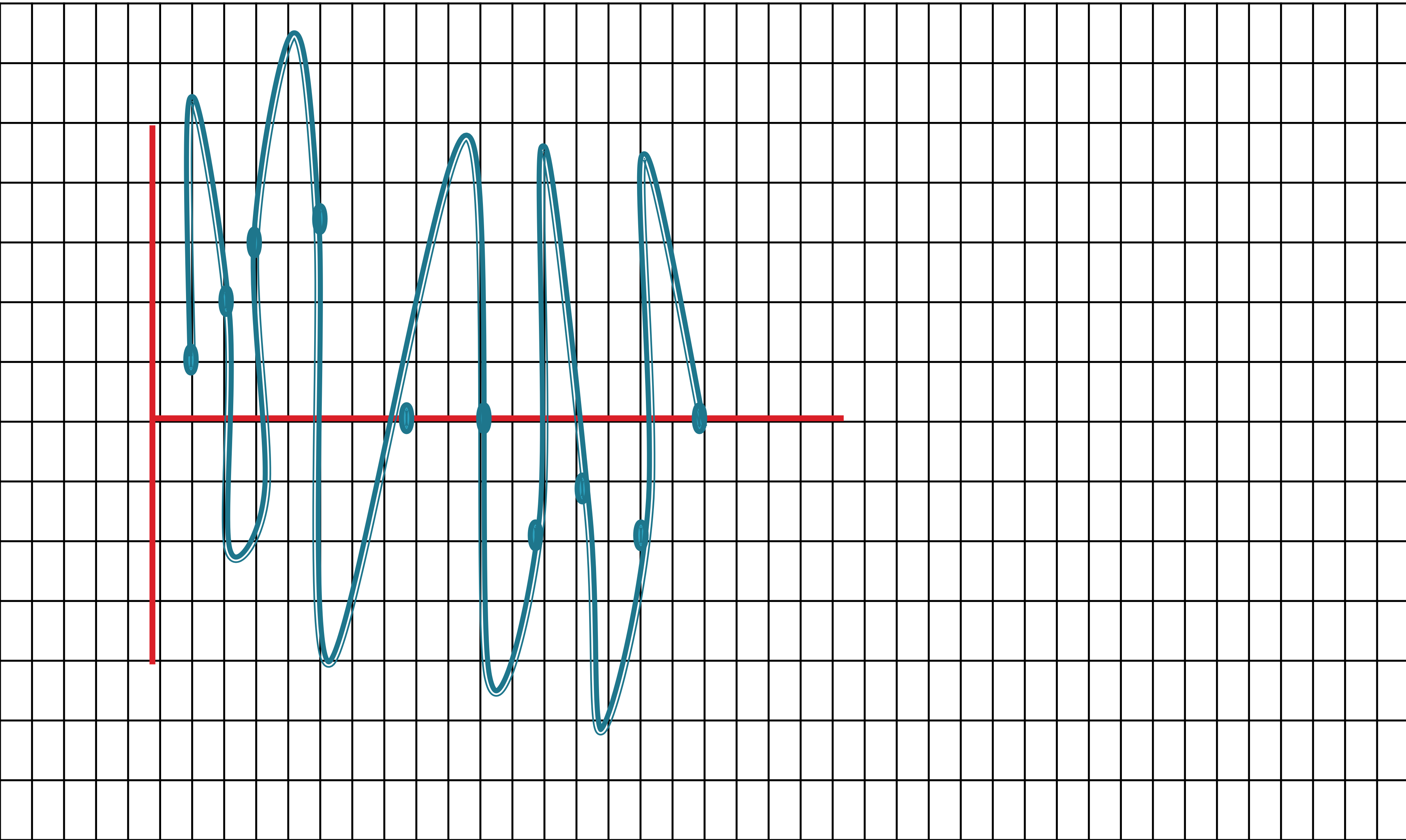
Polynomial Curve Fitting



Polynomial Curve Fitting



Polynomial Curve Fitting



Polynomial Curve Fitting

- Instance Space
- Target Function
- Hypothesis
- Hypothesis set

- Hypothesis should be selected based on
  - $E_{out} = E_{in}$
  - $E_{in} \approx 0$

- Hoeffding Inequality

$$[P(|E_{in}(h) - E_{out}(h))| > \varepsilon] > 2.e^{-2\varepsilon^2 N}$$

Sample Complexity

- The number of examples required for your algorithm to achieve its goals.
- The computational complexity– the resource is CPU cycles.
- In the sample complexity, the resource is labeled examples.

- Hypothesis h is said to best if

$$[P(|E_{in}(h) - E_{out}(h))| > \varepsilon] \leq 4M_{H}\, 2.N.e^{-1/8\varepsilon^{2}N}$$