Lecture 6: Regression Part 1

Leap into the 21st century

ADDO AI

2018

Instructor: Dr. Syed Waqar ul Qounain Jaffry
Assistant Professor PUCIT

AGENDA

- Warmup
  - Introduction
  - Prerequisite Self Check
  - Context Realization
- Linear Regression
  - Introduction
  - Simple Linear Regression
    - Ordinary Least Squares (OLS) Method
    - OLS Analytical Solution
    - Working Example
- Linear Regression (continued)
  - Multi Linear Regression
    - OLS Analytical Solution
    - Gradient Descent Method
    - Working Example
  - Overfitting
  - Regularization and Ridge Regression
  - Conclusion
  - Exercise / Homework
- Class Quiz
- Jupyter Notebook – Lab Work

INTRODUCTION

- To make future predictions based on historic data
- Relationship between various variables
- Requires training dataset
- Results in continuous values
- Multiple forms of regression
  - Simple linear regression
  - Multi-linear regression

TERMINOLOGY

- Input Variables (Independent Variables, Features, Predictors, Covariates)
- Output Variable (Dependent Variable, Response, Target, Criterion, Outcome)
- Base function: $y = f(x) + \epsilon$
- ϵ is irreducible error and part of actual function
  - It is not estimated using regression
- All parameters will be denoted with w followed by their subscript, e.g. w_0 represents 0^{th} parameter of the equation

TERMINOLOGY

- ^ sign shows predicted variables and estimated parameters
- e represents residual error
  - $e = actual \ - predicted = y \ - \hat{y}$
- ϵ is part of residual error and tends to deal with chaos/unpredictability of nature

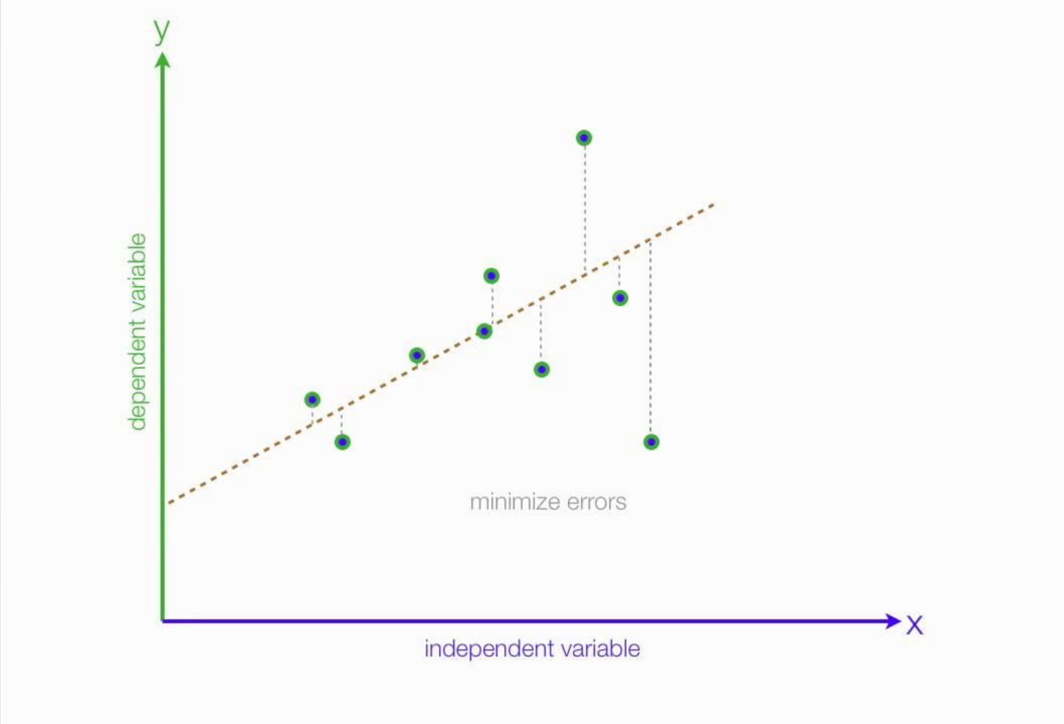
SIMPLE LINEAR REGRESSION

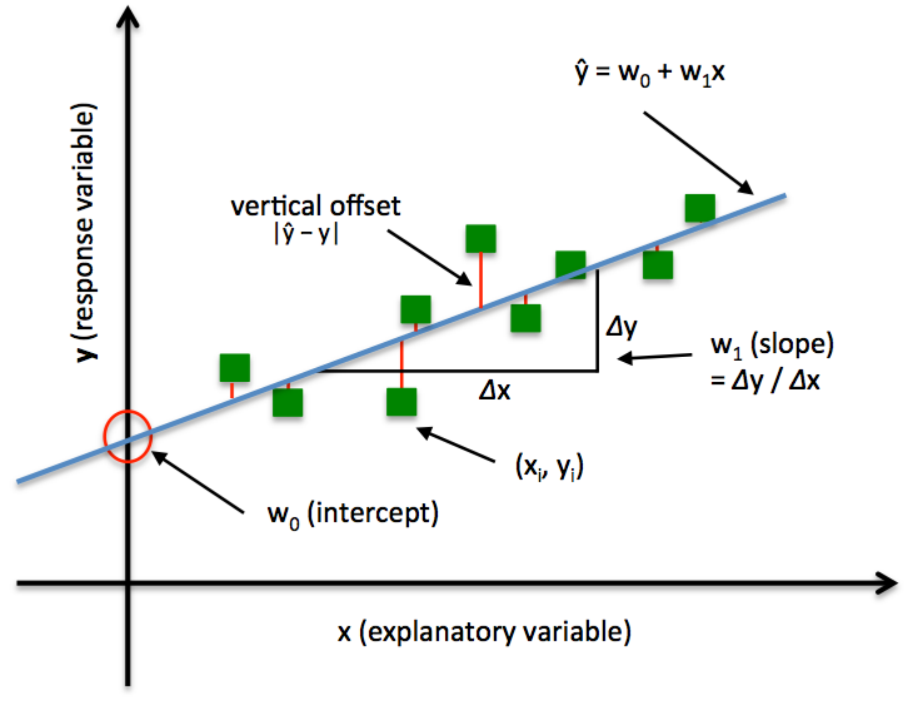
- Also known as Simple regression
- It provides the basics to understand and extend the concept to various other forms of regression
- Applied to find out the impact of one variable to another
- Dependent Variable (y)
- Independent Variable (x)
- Simple linear Equation: $y = f(x) + \epsilon$
  - $f(x) = w_0 + w_1 . x$

SIMPLE LINEAR REGRESSION

- $e = actual\ - predicted = y\ -\ \hat{y}$
  - $e = (w_0 + w_1 . x + \epsilon)\ - (\hat{w}_0 + \hat{w}_1 . x)$
  - $e = (w_0\ - \hat{w}_0) +\ (w_1\ -\ \ \hat{w}_1) . x + \epsilon$
- For any sample i:
  - $e_i = actual\ - predicted = y_i\ -\ \hat{y}_i$
- Goal: To minimize this error against all samples
  - Sum of Squared Error = $Sum\ of\ Squared\ Residuals\ (SSR) = \sum_{i=1}^{n} e_i^2$

SIMPLE LINEAR REGRESSION





SIMPLE LINEAR REGRESSION ANALYTICAL SOLUTION

- Steps
  - Select the regression function
  - Derive the error expression
  - Take its derivative with respect to parameters
  - Set gradient of error function to zero in order to get the optimal value
    - Linear regression is a Convex problem
    - One solution/global minima exists

SIMPLE LINEAR REGRESSION ANALYTICAL SOLUTION

- Goal: Find parameter values that minimize the error.
  - Goal: $argmin_{w_0,w_1} \sum_{i=1}^{n} e_i^2$
  - $argmin_{w_0,w_1} \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = argmin_{w_0,w_1} \sum_{i=1}^{n}\left(y_i - (\hat{w}_0 + \hat{w}_1 . x_i)\right)^2$
  - $argmin_{w_0,w_1} \sum_{i=1}^{n}(y_i - \hat{w}_0 - \hat{w}_1 . x_i)^2$-------------------------------------(1)
- By differentiating eq. (1) w.r.t $\hat{w}_0$:
  - $\frac{\partial E}{\partial \hat{w}_0} = \frac{\partial}{\partial \hat{w}_0} \sum_{i=1}^{n}(y_i - \hat{w}_0 - \hat{w}_1 . x_i)^2 = 2 \sum_{i=1}^{n}(y_i - \hat{w}_0 - \hat{w}_1 . x_i)(-1)$

SIMPLE LINEAR REGRESSION ANALYTICAL SOLUTION

- To find global minima:

$$\frac{\partial E}{\partial \hat{w}_0} = 0 \Rightarrow \; 2\sum_{i=1}^{n}(y_i \; - \; \hat{w}_0 - \hat{w}_1 . x_i)\,(-1) = 0$$

$\Rightarrow \; \sum_{i=1}^{n}(y_i \; - \; \hat{w}_0 - \hat{w}_1 . x_i) \; = 0$ # dropping 2 and -1

$\Rightarrow \; \sum_{i=1}^{n}(y_i \; - \; \hat{w}_1 . x_i) \; = \; \sum_{i=1}^{n}(\hat{w}_0)$

$\Rightarrow \; \sum_{i=1}^{n}(y_i \; - \; \hat{w}_1 . x_i) \; = n\hat{w}_0$ # $\sum_{i=1}^{n}(\hat{w}_0) \; = \hat{w}_0 + \cdots + \; \hat{w}_0 = \; \mathrm{n} * \hat{w}_0$

$$\Rightarrow \; \hat{w}_0 = \frac{\sum_{i=1}^{n}(y_i - \hat{w}_1 . x_i)}{n} = \frac{\sum_{i=1}^{n}(y_i)}{n} - \frac{\sum_{i=1}^{n}(\hat{w}_1 . x_i)}{n}$$

SIMPLE LINEAR REGRESSION ANALYTICAL SOLUTION

$$\Rightarrow \hat{w}_0 = \frac{\sum_{i=1}^{n}(y_i\ )}{n} - \hat{w}_1 \frac{\sum_{i=1}^{n}(x_i)}{n}$$

$$\Rightarrow \ \hat{w}_0 = Average(y) - \hat{w}_1 Average(x)$$

$$\Rightarrow \ \hat{\mathbf{w}}_{\mathbf{0}} = \ \overline{\boldsymbol{y}} - \hat{\mathbf{w}}_{\mathbf{1}}\overline{\boldsymbol{x}}$$

- By differentiating eq. (1) w.r.t $\hat{w}_1$:

$$\frac{\partial E}{\partial \hat{w}_1} = \frac{\partial}{\partial \hat{w}_1}\sum_{i=1}^{n}(y_i\ -\ \hat{w}_0 - \hat{w}_1 . x_i)^2 = \ 2\ \sum_{i=1}^{n}(y_i\ -\ \hat{w}_0 - \hat{w}_1 . x)\ (-x_i)$$

SIMPLE LINEAR REGRESSION ANALYTICAL SOLUTION

- By substituting the value of $\hat{w}_0$ in gradient equation and equating it with zero the value of $\hat{w}_1$ becomes

$$\hat{w}_1 = \frac{\sum_{i=1}^{n} x_i y_i - \frac{1}{n}\sum_{i=1}^{n} y_i \sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} x_i^2 - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)^2} = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{y}\,\bar{x}}{\sum_{i=1}^{n} x_i^2 - n\overline{x^2}}$$

$$\hat{w}_0 = \bar{y} - \hat{w}_1 \bar{x}\,, \quad \hat{w}_1 = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{y}\,\bar{x}}{\sum_{i=1}^{n} x_i^2 - n\overline{x^2}}$$

- Using these two equations, one can estimate parameter values for any simple linear regression problem.

SIMPLE LINEAR REGRESSION WORKING EXAMPLE

- x presents the house area and
- y presents house price in millions
- Using analytical solution
  - Both parameters can be found
  - Assuming no irreducible error (ϵ)
- Final Equation:
  - $\hat{y} = -7.96 + 0.19\,x$
- To predict y for x = 64 is
  - $\hat{y} = -7.96 + 0.19\,(64)$
  - $\hat{y} = 4.2$
- Residual error e for x = 63 is
  - $\hat{y} = -7.96 + 0.19\,(63)$
  - $\hat{y} = 4.2$
  - $e = y - \hat{y} \quad = 4 - 4.01 = 0.01$

	$\mathbf{x}$	$\mathbf{y}$	$\mathbf{xy}$	$\mathbf{x^2}$
n = 5	60	3.1	186	3600
	61	3.6	219.6	3721
	62	3.8	235.6	3844
	63	4	252	3969
	65	4.1	266.5	4225
Sum	311	18.6	1159.7	19359
Average	62.2	3.72	231.94	3868.84
Slope ($\hat{w}_1$)	0.187837	0.19*	* Represents rounded off numbers	
Intercept ($\hat{w}_0$)	-7.96351	-7.96*		

MULTIPLE LINEAR REGRESSION MOTIVATION

- Simple linear regression
  - Only applicable when there is only one input variable
- Real life problems
  - Carry multiple input factors
- Solution
  - Generalization of simple linear regression

MULTIPLE LINEAR REGRESSION

- A single training instance will form a vector
- Thus, represent complete input in the form of a matrix
- Transform parameters in vector
- Example
  - Age, Experience, Salary as input variable
  - Salary Bonus as output variable
  - Salary Bonus $= \hat{w}_{\mathbf{0}} + \hat{w}_{\mathbf{1}} * \boldsymbol{Age} + \hat{w}_{\mathbf{2}} * \boldsymbol{Experience} + \hat{w}_{\mathbf{3}} * \boldsymbol{Salary} + \epsilon$

MULTIPLE LINEAR REGRESSION

- Resulting Vectors
- N: total data points M: total input dimensions
  - $\vec{\hat{y}}$ $(predicted\ output\ vector)$ size: N * 1
  - $\vec{\hat{w}}$ $(parameter\ vector)$ size: (M +1) $*$ 1 # to incorporate $\hat{w}_0$
  - $\tilde{X}$ $(Input\ Matrix)$ size: N * (M + 1) # to incorporate $\hat{w}_0$
  - $\vec{y}$ $(actual\ output\ vector)$ size: N * 1
  - $\vec{\epsilon}$ $(\ irreducible\ error\ vector)$ size: N * 1

MULTIPLE LINEAR REGRESSION

- Resulting Vectors
  - $\vec{y} = \begin{bmatrix} 2000 \\ 2250 \\ 2600 \\ 2350 \\ 1850 \end{bmatrix}$ $\qquad \vec{\hat{w}} = \begin{bmatrix} \hat{w}_0 \\ \hat{w}_1 \\ \hat{w}_2 \\ \hat{w}_3 \end{bmatrix}$
  - $\tilde{X} = \begin{bmatrix} 1 & 25 & 1 & 40 \\ 1 & 30 & 2 & 50 \\ 1 & 35 & 9 & 65 \\ 1 & 33 & 4 & 55 \\ 1 & 23 & 0 & 35 \end{bmatrix}$

X			Y
Age	Exp.	Salary(K)	Bonus
25	1	40	2000
30	2	50	2250
35	9	65	2600
33	4	55	2350
23	0	35	1850

  - In input matrix; 1st column containing 1's is added to incorporate bias effect

MULTIPLE LINEAR REGRESSION

- By performing $\tilde{X}\vec{\hat{w}}$: we get

$$\vec{\hat{y}} = \begin{matrix} \hat{w}_0 + \hat{w}_1 * \mathbf{25} + \hat{w}_2 * \mathbf{1} + \hat{w}_3 * \mathbf{40} \\ \hat{w}_0 + \hat{w}_1 * \mathbf{30} + \hat{w}_2 * \mathbf{2} + \hat{w}_3 * \mathbf{50} \\ \hat{w}_0 + \hat{w}_1 * \mathbf{35} + \hat{w}_2 * \mathbf{9} + \hat{w}_3 * \mathbf{65} \\ \hat{w}_0 + \hat{w}_1 * \mathbf{33} + \hat{w}_2 * \mathbf{4} + \hat{w}_3 * \mathbf{55} \\ \hat{w}_0 + \hat{w}_1 * \mathbf{23} + \hat{w}_2 * \mathbf{0} + \hat{w}_3 * \mathbf{35} \end{matrix}$$

MULTIPLE LINEAR REGRESSION ANALYTICAL SOLUTION

- Error can then be expressed as:
  - error = $\vec{y} - \vec{\hat{y}} = \vec{y} - \tilde{X}\overrightarrow{\hat{w}}$
  - Sum of residual error $\sum_{i=1}^{n} e_i^2$ in matrix form can the be expressed as: $e'e$
  - Error function (E) $= \left(\vec{y} - \tilde{X}\overrightarrow{\hat{w}}\right)^2 = \left(\vec{y} - \tilde{X}\overrightarrow{\hat{w}}\right)'\left(\vec{y} - \tilde{X}\overrightarrow{\hat{w}}\right)$ # $'$ represents transpose
  - Error function (E) = $\left(\vec{y}' - \overrightarrow{\hat{w}}'\tilde{X}'\right)\left(\vec{y} - \tilde{X}\overrightarrow{\hat{w}}\right)$ # $(AB)' = B'A'$
  - E $= \vec{y}'\vec{y} - \vec{y}'\tilde{X}\overrightarrow{\hat{w}} - \overrightarrow{\hat{w}}'\tilde{X}'\vec{y} + \overrightarrow{\hat{w}}'\tilde{X}'\tilde{X}\overrightarrow{\hat{w}}$

MULTIPLE LINEAR REGRESSION ANALYTICAL SOLUTION

- Compute gradient of Error function with respect to parameters
- Set it equal to zero to get solution

$$\frac{\partial E}{\partial \vec{\hat{w}}} = \frac{\partial}{\partial \vec{\hat{w}}} \overrightarrow{(y}'\vec{y} - \vec{y}'\tilde{X}\vec{\hat{w}} - \vec{\hat{w}}'\tilde{X}'\vec{y} + \vec{\hat{w}}'\tilde{X}'\tilde{X}\vec{\hat{w}}) = \vec{0}$$

$$\Rightarrow (0 - (\vec{y}'\widetilde{X)})' - \tilde{X}'\vec{y} + 2\tilde{X}'\tilde{X}\vec{\hat{w}}) = \vec{0}$$

$$\Rightarrow -2\tilde{X}'\vec{y} + 2\tilde{X}'\tilde{X}\vec{\hat{w}} = \vec{0}$$

$$\Rightarrow \tilde{X}'\tilde{X}\vec{\hat{w}} = \overrightarrow{\tilde{X}'\vec{y}}$$

- Solving it generates solution for

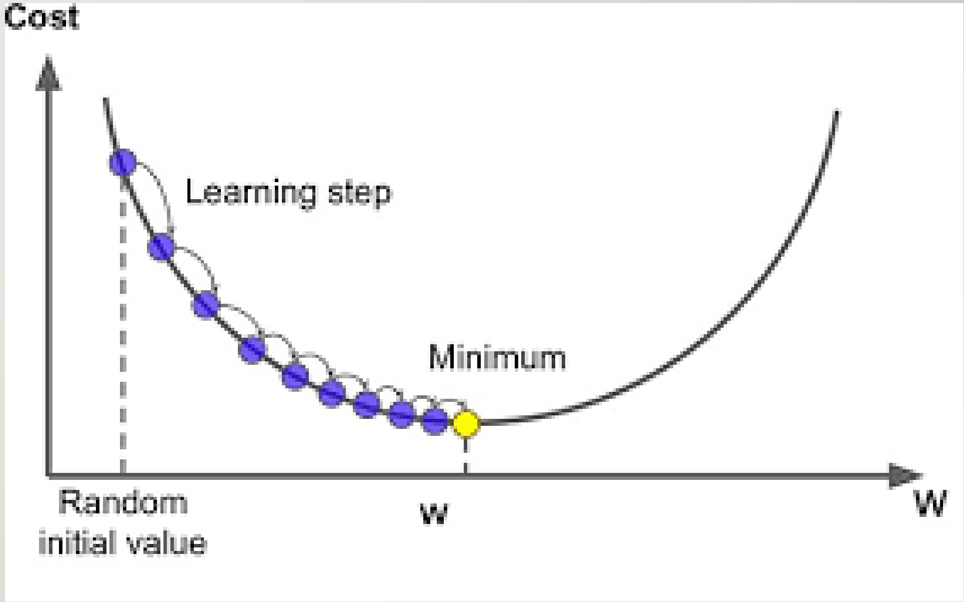
$$\vec{\hat{w}} = (\boldsymbol{\widetilde{X}'\widetilde{X}})^{-\mathbf{1}}\,\boldsymbol{\widetilde{X}'\vec{y}}$$

ORDINARY LEAST SQUARES

- Ordinary least squares (OLS)
  - Provides analytical solution
  - Both methods described are OLS
- Issues with OLS
  - Requires invertible matrix
  - For huge data set, matrix inversion gets very time-consuming

GRADIENT DESCENT

- Alternate way to train various ML models
- Regression problems
  - One global minima
- Required
  - Learning rate/step size
  - Gradient function
- After every step:
  - $weight_{new} = weight_{old} - \ learning\ step\ *\ gradient$



GRADIENT DESCENT WORKING EXAMPLE

For Simple Linear regression, gradients are:

$$\frac{\partial E}{\partial \hat{w}_0} = \frac{\partial}{\partial \hat{w}_0} \sum_{i=1}^{n} (y_i - \hat{w}_0 - \hat{w}_1 . x_i)^2 = 2 \sum_{i=1}^{n} (y_i - \hat{y})(-1)$$

$$\frac{\partial E}{\partial \hat{w}_1} = \frac{\partial}{\partial \hat{w}_1} \sum_{i=1}^{n} (y_i - \hat{w}_0 - \hat{w}_1 . x_i)^2 = 2 \sum_{i=1}^{n} (y_i - \hat{y})(-x_i)$$

Assign random values parameters $\hat{w}_0 = 0.5$, $\hat{w}_1 = 0.25$ learning rate = 0.25

Compute $\hat{y} = 0.25 * 60 + 0.5 = 15.5 \Rightarrow \frac{\partial E}{\partial \hat{w}_0} = 2 * (3.1 - 15.5)(-1) = 24.8$

$$\frac{\partial E}{\partial \hat{w}_1} = 2 * (3.1 - 15.5)(-1)(60) = 1448$$

Hence $\hat{w}_0 = 0.5 – (0.25)*24.8 = -5.7$, $\hat{w}_1 = 0.25 – (0.25)*1448 = -361.75$

x	y
60	3.1
61	3.6
62	3.8
63	4
65	4.1

GRADIENT DESCENT MULTILINEAR REGRESSION

For Multiple Linear regression, we can use general notation:

Error = $\vec{y} - \vec{\hat{y}} = \vec{y} - \tilde{X}\vec{\hat{w}}$

$$\text{SSR} = \text{E} = \left(\vec{y} - \tilde{X}\vec{\hat{w}}\right)^2 = \left(\vec{y} - \tilde{X}\vec{\hat{w}}\right)'\left(\vec{y} - \tilde{X}\vec{\hat{w}}\right)$$

$$E = \left(\vec{y}' - \vec{\hat{w}}'\tilde{X}'\right)\left(\vec{y} - \tilde{X}\vec{\hat{w}}\right) = \vec{y}'\vec{y} - \vec{y}'\tilde{X}\vec{\hat{w}} - \vec{\hat{w}}'\tilde{X}'\vec{y} + \vec{\hat{w}}'\tilde{X}'\tilde{X}\vec{\hat{w}}$$

$$\overrightarrow{Gradient} = \frac{\partial E}{\partial \vec{\hat{w}}} = -(\widetilde{\vec{y}'X})' - \tilde{X}'\vec{y} + 2\tilde{X}'\tilde{X}\vec{\hat{w}} = -2\tilde{X}'\vec{y} + 2\tilde{X}'\tilde{X}\vec{\hat{w}}$$

$$\frac{\partial E}{\partial \vec{\hat{w}}} = -2 * \widetilde{X'} * \left(\vec{y} - \widetilde{X}\,\vec{\hat{w}}\right)$$

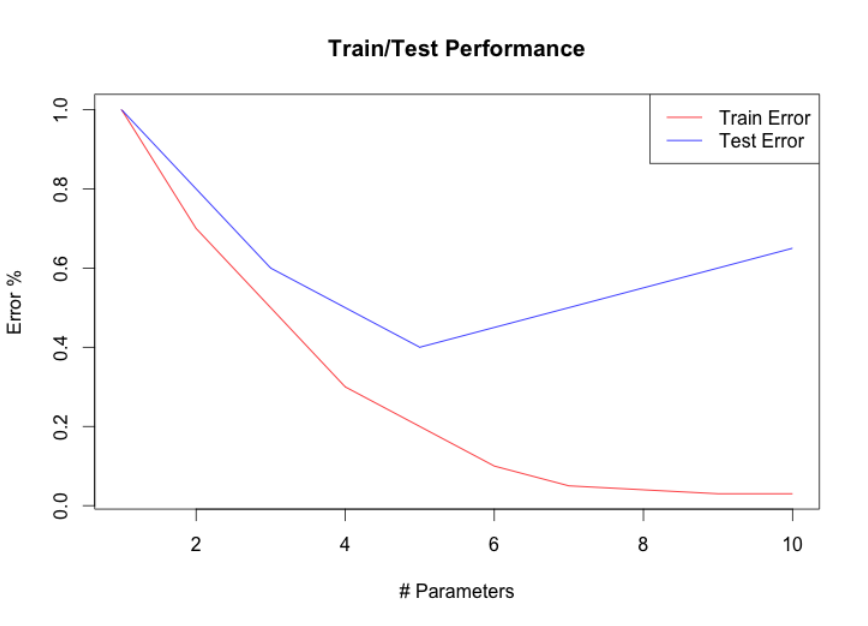
$$\overrightarrow{\hat{w}_{new}} = \overrightarrow{\hat{w}_{old}} - learning\ step * \overrightarrow{Gradient}/N$$

VARIOUS TYPES OF GRADIENT DESCENTS

- Simple Gradient Descent
  - It refers to gradient descent after whole dataset iteration
- Stochastic
  - It computes gradient descent after every example; as show in previous example
- Batch
  - In this mode, batches are made, and weights are updated using whole batch as input
- Normally, stochastic batch gradient descent is used.

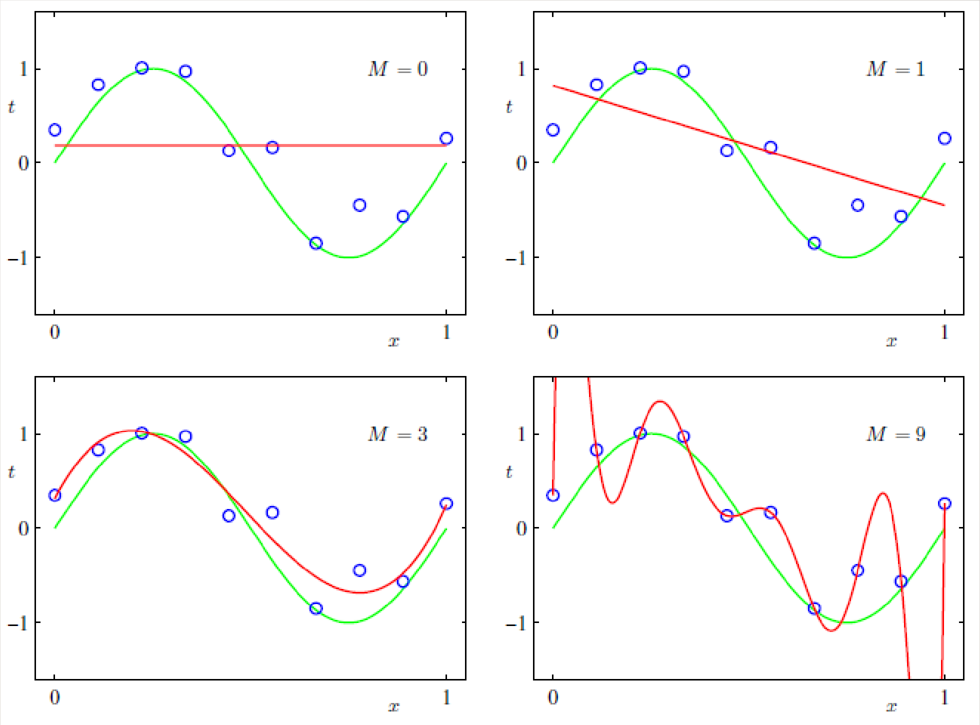
LINEAR REGRESSION OVERFITTING

- Dataset division
  - Training set
  - Test set
- Aim is to reduce training error as well as validation error
- If $train_{error} \ll test_{error}$
  - Overfitting occurred
- else
  - Continue training



LINEAR REGRESSION OVERFITTING

- M refers to the no. of input dimensions
- Various models having various powers
- Increase in dimensions
  - More power in model
  - More parameters to tune
  - Result in overfitting (last figure with M=9)
- Solution: Regularization



LINEAR REGRESSION REGULARIZATION

- Regularization
  - Penalizes on the basis of parameter's magnitude
  - Helps in generalization (avoids overfitting)
- Two widely used schemes:
  - L2-norm/ Ridge Regression (Squared Norm)
  - L1-norm (Absolute Value)

REGULARIZATION
HOW TO DO IT ?

- In order to perform Regularization : error function gets updated
  - $SSR = \sum_{i=1}^{n} e_i^2$ *(without regularization),*
  - $SSR = \sum_{i=1}^{n} e_i^2 + \frac{\lambda}{2} \|\hat{w}\|^2$ *(with regularization)*
  - $\frac{\partial E}{\partial \hat{w}_i} = -2 \sum_{i=1}^{n} e_i \quad * x_i + \frac{\partial E}{\partial \hat{w}_i} (\frac{\lambda}{2} (w_0^2 + \cdots + w_i^2 + \cdots + w_n^2))$
  - $\frac{\partial E}{\partial \hat{w}_i} = -2 \sum_{i=1}^{n} e_i \quad * x_i + \frac{\lambda}{2} * \frac{\partial E}{\partial \hat{w}_i} ((w_0^2 + \cdots + w_i^2 + \cdots + w_n^2))$
  - $\frac{\partial E}{\partial \hat{w}_i} = -2 \sum_{i=1}^{n} e_i \quad * x_i + \frac{\lambda}{2} * \ ((0 + \cdots + 2 * w_i \ + \cdots + 0))$
  - $\frac{\partial E}{\partial \hat{w}_i} = -2 \sum_{i=1}^{n} e_i \quad * x_i + \frac{\lambda}{2} * \ (2 * w_i \ )$
  - $\frac{\partial E}{\partial \hat{w}_i} = -2 \sum_{i=1}^{n} e_i \quad * x_i + \lambda w_i$

REGULARIZATION
REVISED ERROR FUNCTION

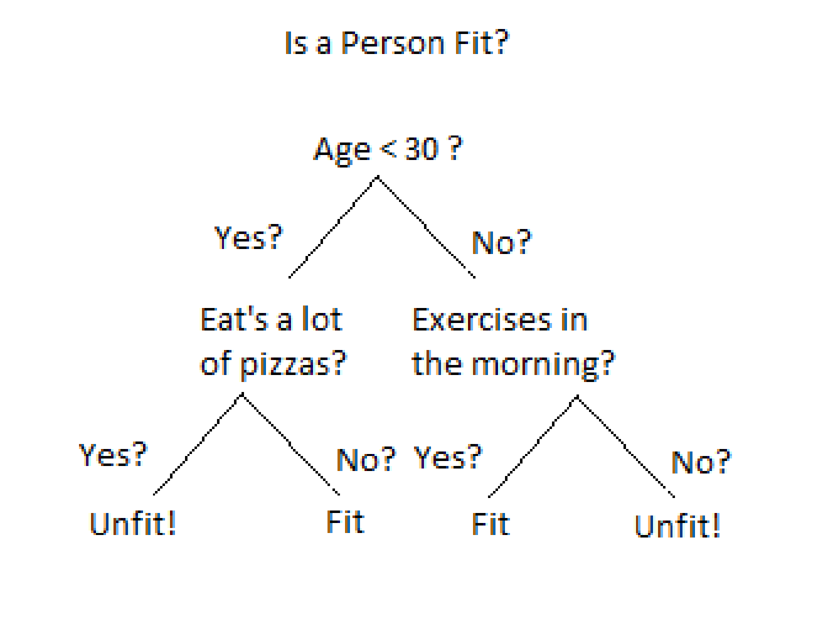
- In case of Multiple linear regression gradient function becomes
  - $$\frac{\partial E}{\partial \hat{w}_i} = -2\sum_{i=1}^{n} e_i \quad * x_i + \lambda w_i$$ ---------------------------(2)
- In case of Multiple linear Ridge regression, optimal solution becomes
  - $$\vec{\hat{w}} = (\widetilde{\boldsymbol{X}}'\widetilde{\boldsymbol{X}} + \lambda\mathbf{I})^{-1}\,\widetilde{\boldsymbol{X}}'\vec{\boldsymbol{y}}$$ ------------------------------------(3) # I is identity matrix
  - *Highlighted segment in red is known as L2-norm*
  - *Particular case of L2-norm presented in Eq. 2 and 3 is called Ridge regression*

LINEAR REGRESSION CONCLUSION

- So far, regression studied results into continuous variable and is only applicable if function is linear in terms of input variables
- If input parameters are non-linear, then kernel trick can be used to transform input space into a linear space
- Kernel trick refers to transformation of data from one space to another.
  - E.g. I have input having 10 parameters, by applying PCA, I can transform it in 3 parameters space
  - This transformation from one space to another using some transformation function, is regarded as kernel trick

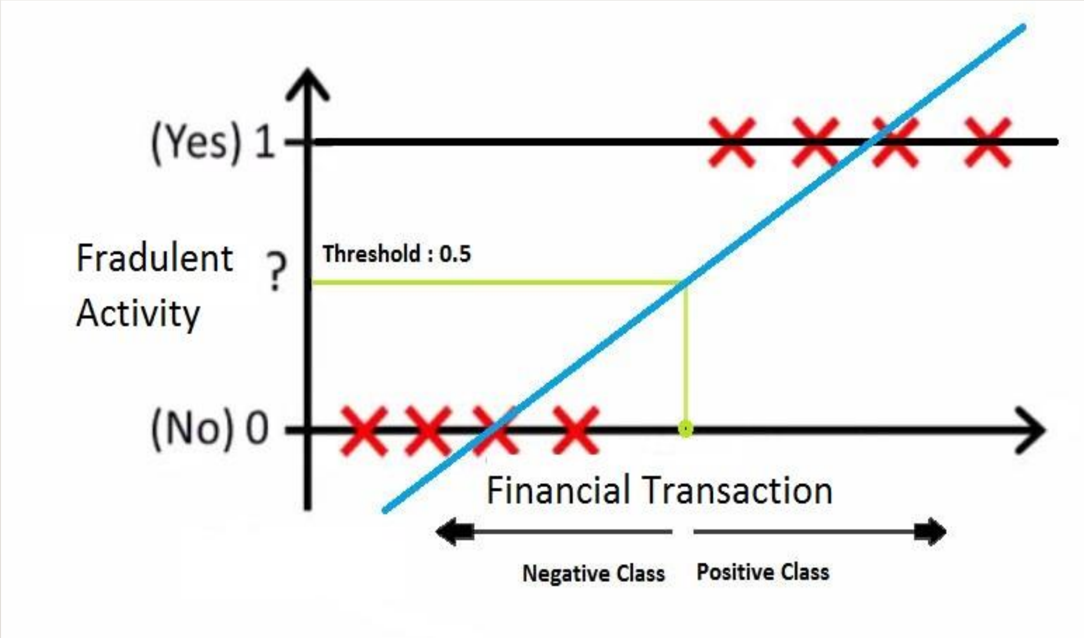
CLASSIFICATION

- Predict discrete rather then continuous dependent variable



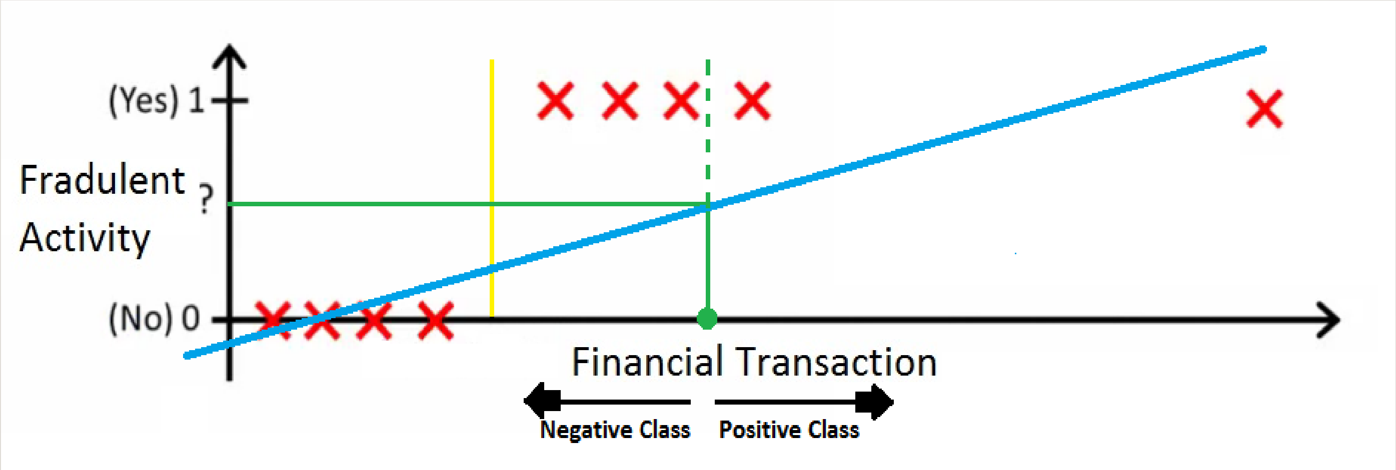
WHY NOT USE LINEAR REGRESSION FOR CLASSIFICATION

- Linear regression finds out the linear model that best fits the data
- It results in continuous values
- Threshold can be used for classification



WHY NOT USE LINEAR REGRESSION FOR CLASSIFICATION

- Generalization is challenging
- Boundary between classes can get confused
- Level of certainty on results can't be acquired
- Solution: Logistic Regression



LOGISTIC REGRESSION

- Logistic regression is used for classification
  - Binary classification (Exactly two classes)
  - Multi-class classification (More than two classes)
- There are many phenomenon that require binary decision
  - Fraud detection
  - Gender prediction
  - Tumor classification
- In next lecture, we'll cover logistic regression and remaining concepts