Data Science & Machine Learning
: Regression and Multi-objective Optimization

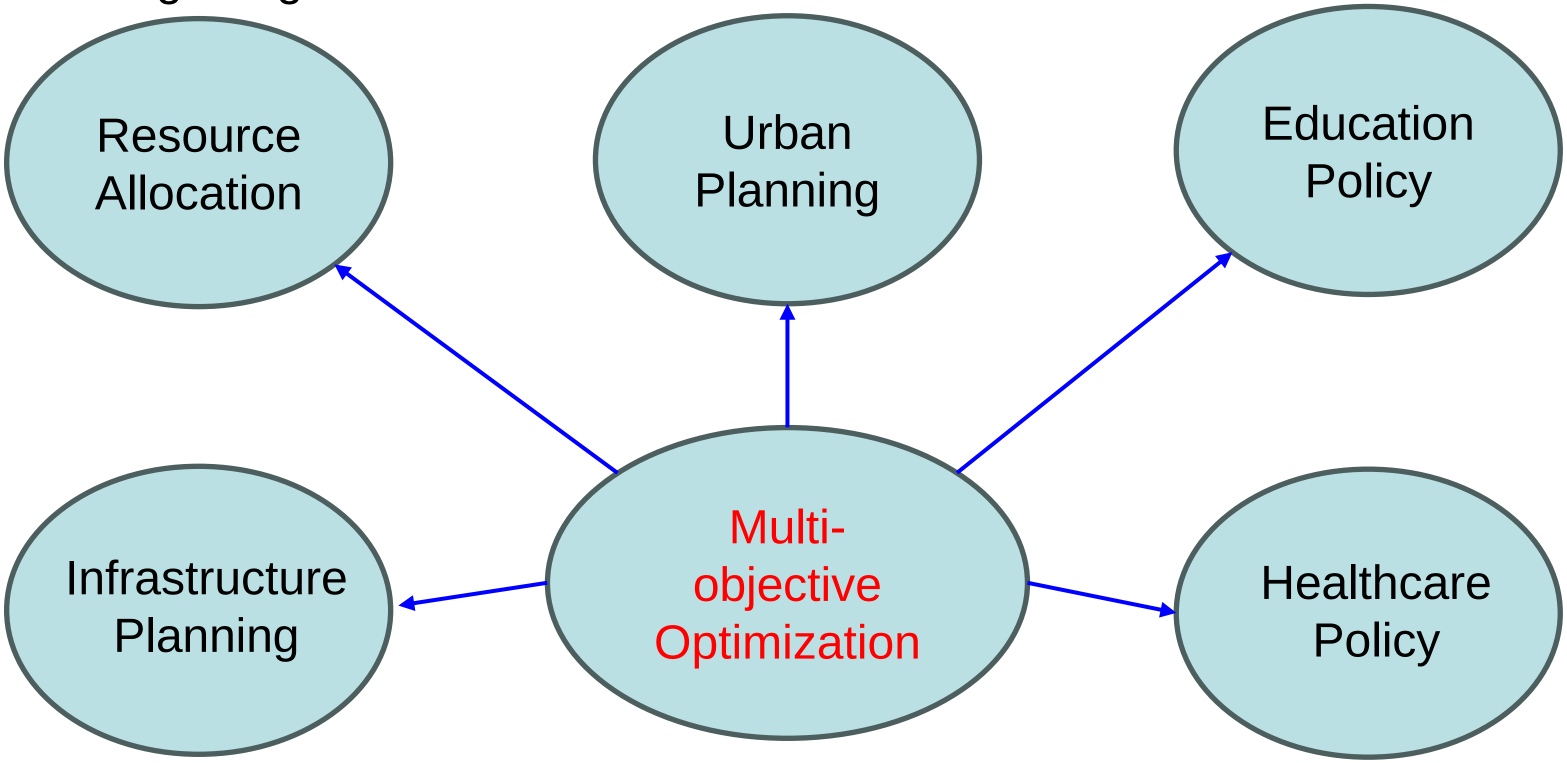
Dr. Manojkumar C. Ramteke, Dr. Hariprasad Kodamana, Dr. Agam Gupta

Department of Chemical Engineering

IIT Delhi

28th February, 2024

Application of Multi-objective Optimization



Resource Allocation

Maximizing economic growth, minimizing poverty, and promoting environmental sustainability when allocating budgets

Urban Planning

Transportation efficiency, housing affordability, green space provision, and air quality improvement

Education Policy

Allocate funding to schools based on criteria such as student performance, socio-economic background, and geographic location

Multi-objective Optimization

Infrastructure Planning

Cost-effectiveness, environmental impact, and social equity

Healthcare Policy

Maximize population health outcomes while minimizing disparities in access to care

Case Study

Background: Portfolio optimization is a financial concept that involves selecting the best combination of assets for investment, aiming to maximize returns while minimizing risk. It's commonly associated with investment professionals, fund managers, and individual investors seeking to build wealth or achieve specific financial goals. However, the relevance of portfolio optimization extends beyond the realm of finance and can indeed be applicable to bureaucrats and policymakers in various ways such as Public Funds Management, Risk Management, Infrastructure Investments, and Pension Fund Management.

In this case study, a dataset involving 32 years of data on the return rate of different stocks (T. Bills, T. Bonds, NASDAQ, Dow Jones, S&P500, Gold) is given.

SR. No	Year	T. Bills	T. Bonds	NASDAQ	Dow Jones	S&P500	Gold
1	1980.0	1.1122	0.9701	1.3388	1.1493	1.2577	1.208
2	1981.0	1.1430	1.0820	0.9679	0.9077	0.9027	0.746
3	1982.0	1.1101	1.3281	1.1867	1.1961	1.1476	1.083
4	1983.0	1.0845	1.0320	1.1987	1.2027	1.1727	0.876
5	1984.0	1.0961	1.1373	0.8878	0.9626	1.0140	0.822
6	1985.0	1.0749	1.2571	1.3136	1.2766	1.2633	1.002

Case Study

Problem: For the given data on year-wise return rates of six different stocks, perform the multi-objective optimization to invest INR 100.

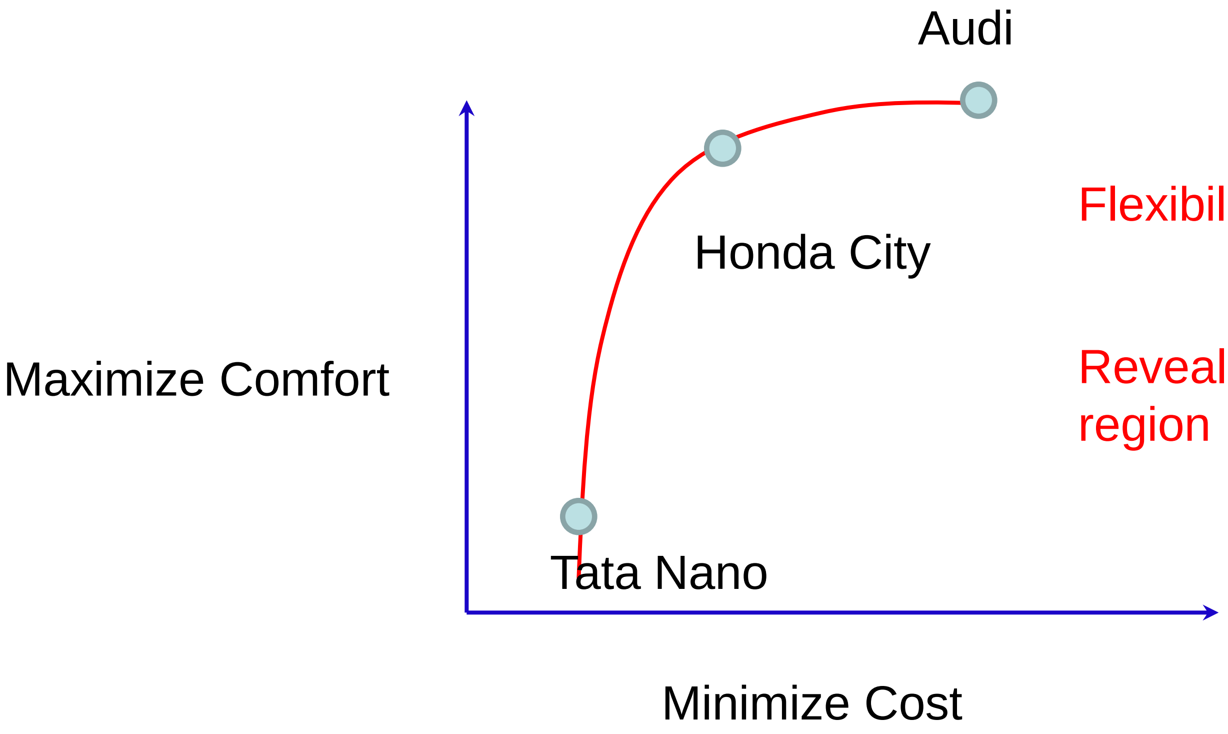
1) Perform multi-objective optimization to maximize return on investment and minimize the risk associated with the given stocks using a non-dominated sorting genetic algorithm.
2) Analyze the Pareto optimal results obtained.
3) Select the suitable operating point on the Pareto optimal front for optimal investment which not only gives reasonable return but also keeps the risk low.

AIM: To understand the concept of multi-objective optimization and genetic algorithm.

Applications:

The real-life problems often involve multiple objectives to satisfy. Particularly, decision-making always involves multiple objectives to satisfy. In such a situation, obtaining a complete trade-off is extremely useful as it can enrich decision-making. A typical example is to select infrastructural projects that not only improve the economic and social benefits but also minimize environmental degradation.

Why Multi-Objective Optimization?



Flexibility of multiple solutions

Reveals the best operating region

While purchasing a car, a customer has two things in mind: **Maximize Comfort** and **Minimize Cost**

Generalized Optimization Formulation

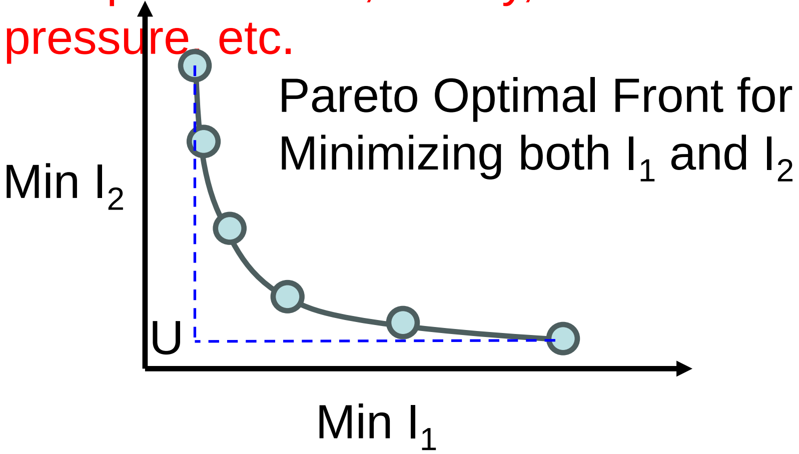
Objective Functions: Multiple objectives (I_1, I_2, etc.) are important in real life situations (e.g., maximising economic benefits, minimizing environmental degradation, minimizing adverse social impacts is simultaneously required in government projects such as Highways, and other construction projects).

Inequality Constraints: | Equality Constraints:

These are selected based on various restrictions based on safety, design, operational practices or social requirements. This includes available technology, skills, labor, budget, construction tolerance, completion date, safety, local climatic conditions, regulation, public concern, media pressure, etc.

Bounds:

These are selected based on the given operating ranges of the key design variables

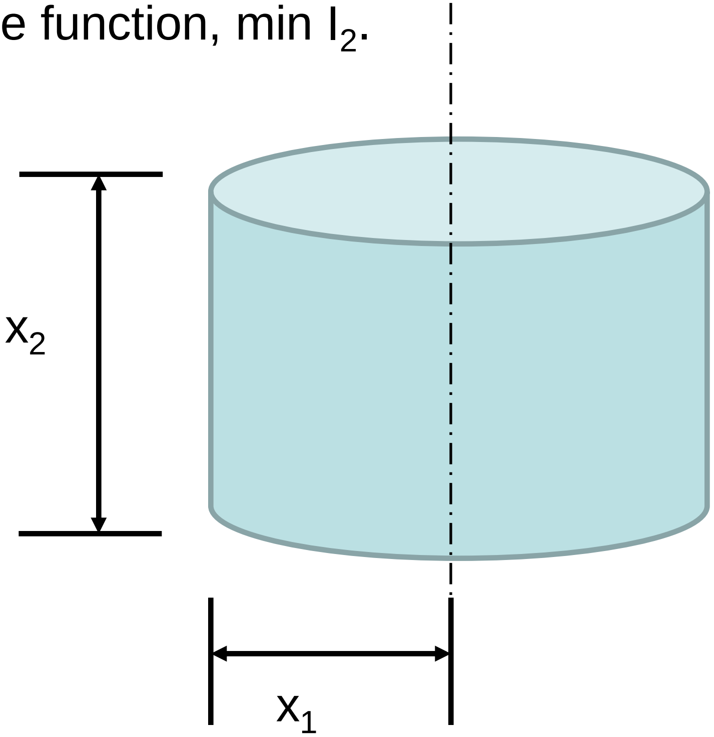


Pareto Optimal Front for Minimizing both I_1 and I_2

1) In single objective optimization, where number of objectives (m) = 1. Usually, single optimal solution is obtained.
2) In multi-objective optimization, where number of objectives (m) > 1. Typically, multiple equally good solutions are obtained. All these solutions are referred as Pareto optimal solutions. Decision maker selects one of these as operating solution based on his experience.

Simple Example

Example: To illustrate the concept of multi-objective optimization, consider a simple example of a cylindrical container with radius, x_1 and height x_2 as shown in Figure. A possible objective function, min I_1, is to minimize the total surface area of the cylindrical container having a specified volume (a constraint). This is meaningful since it leads to the least amount of sheet-metal for construction. However, for pressure stability, the lateral surface area is more important and, therefore, another meaningful objective function, min I_2.



Problem Formulation

MOO

Objective functions:

$$\min I_1(x_1, x_2) \equiv 2\pi x_1 (x_1 + x_2)$$

$$\min I_2(x_1, x_2) \equiv 2\pi x_1 x_2$$

SOO 1

Objective function:

$$\min I_1(x_1, x_2) \equiv 2\pi x_1 (x_1 + x_2)$$

SOO 2

Objective function:

$$\min I_2(x_1, x_2) \equiv 2\pi x_1 x_2$$

Bounds:

$$0 \le x_1 \le 10 \text{ m}$$

$$0 \le x_2 \le 10 \text{ m}$$

Constraint:

$$g(\boldsymbol{x}) = \pi x_1^2 x_2 = 100 \text{ m}^3$$

Replace x_2 in terms of x_1 using the equality constraint

$$x_2 = \frac{100}{\pi x_1^2}$$

Solution of SOO

SOO 1

$$\min I_1(x_1) \equiv 2\pi x_1\left(x_1+\frac{100}{\pi x_1^2}\right)=2\left(\pi x_1^2+\frac{100}{x_1}\right)$$

The analytical solution is obtained as:

$$\frac{dI_1}{dx_1}=2\pi x_1-\frac{100}{x_1^2}=0\Rightarrow x_1^{opt}=2.51m\Rightarrow x_2^{opt}=5.03m$$

$\frac{d^2I_1}{dx_1^2}=$ $2\pi+200/x_1^3$ optimum obtained is a minimum

The optimum is [2.51, 5.03] which leads to a minimum total surface area of 118.85 m^2 but lateral surface area = 80 m^2.

SOO 2

$$\min I_2(x_1)\equiv 2\pi x_1\left(\frac{100}{\pi x_1^2}\right)=\frac{200}{x_1}$$

$$\frac{dI_2}{dx_1}=-\frac{200}{x_1^2}=0\Rightarrow x_1^{opt}=\infty\Rightarrow x_1^{opt}=10\text{ m}$$

$$(\text{upper bound})\Rightarrow x_2^{opt}=0.32\text{ m}$$

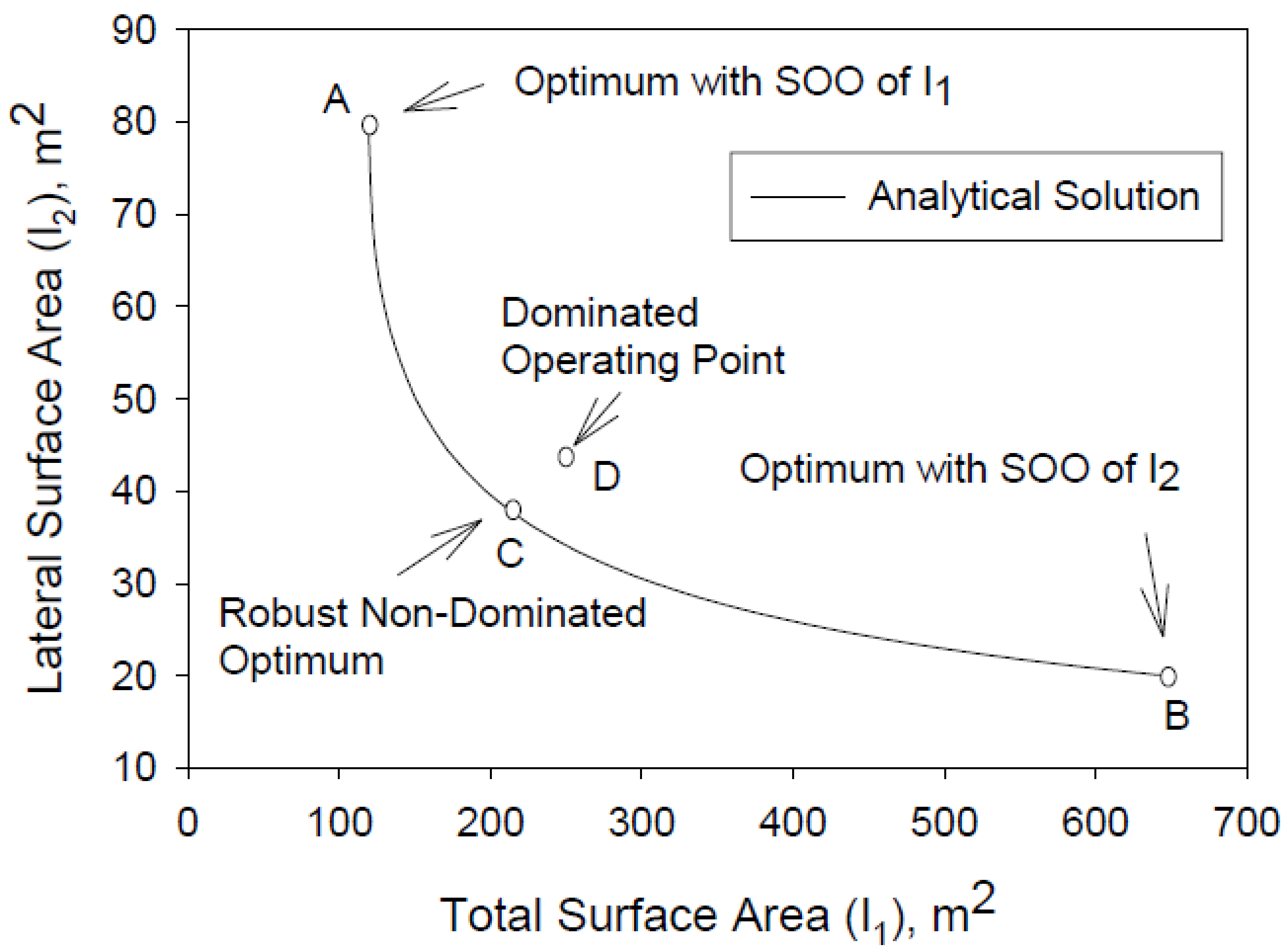
$\frac{d^2I_2}{dx_1^2}=\frac{400}{x_1^3}\Big|_{x_1^{opt}=10\text{ m}}=\frac{400}{1000}=0.4>0\Rightarrow$ the

the optimum obtained is a minimum

The optimum is [10, 0.32] which leads to a minimum lateral surface area = 20 m^2 but total surface area of 648 m^2.

These optimum of single objective optimization represents the corner solutions in multi-objective optimization

Solution of MOO



Such representation of results is referred as Pareto optimal plot

Conventional Multi-Objective Optimization Techniques

1) Goal Programming: The difference of different objectives from set target values is summed up for all objectives and this sum is minimized in a single objective optimization.
2) Epsilon Constraint Method: Only one objective is optimized while constraining the remaining objectives to different values.
3) Utility Function method (Weighted Sum Method): All objectives are summed up in a weighted manner and the sum is optimized.
4) Lexicographic Approach: The objectives are ranked based on their importance. The most important objective is optimized first. Then this objective is constrained to optimal value, and the next objective in the ranking is optimized.

These provides single optimal solution at a time and to generate the entire Pareto optimal front optimization must be performed repeatedly.

Metaheuristic Multi-Objective Optimization Techniques

1) Popular metaheuristic algorithms such as genetic algorithm, differential evolution, particle swarm optimization, etc. can solve multi-objective, formulations and can generate the compete Pareto front in one run.

Weighted Sum Method (Hand Calculation)

Summation of objectives $I_1, I_2, \dots, I_m$:

I = Weight 1 × Objective 1 + Weight 2 × Objective 2 +…+ Weight m × Objective m

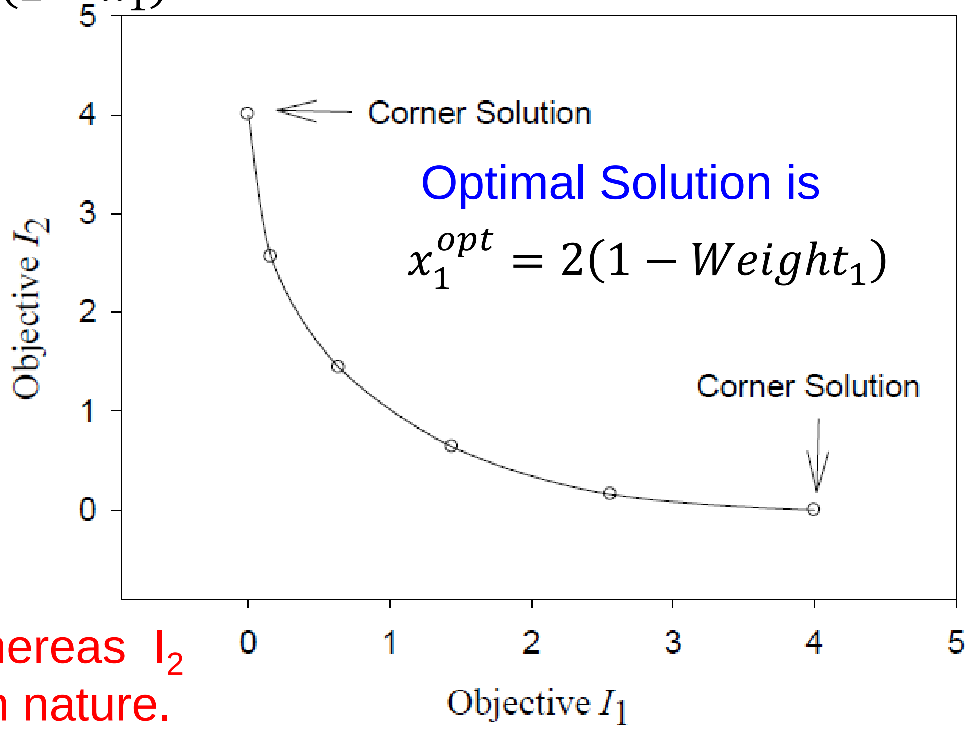
Summation of weights should be equals to 1: Weight 1 + Weight 2 +…+ Weight m = 1

Solve the following Example with weights varying in the interval of 0.2

Objective 1: $\min I_1 = x_1^2$ Objective 2: $\min I_2 = (2 - x_1)^2$ Bounds: $x_1 \in [0,2]$

$$\min I = Weight_1 \times x_1^2 + (1 - Weight_1) \times (2 - x_1)^2$$

$Weight_1$	x_1^{opt}	I_1^{opt}	I_2^{opt}
0	2	4	0
0.2	1.6	2.56	0.16
0.4	1.2	1.44	0.64
0.6	0.8	0.64	1.44
0.8	0.4	0.16	2.56
1.0	0	0	4



Optimal Solution is

$$x_1^{opt} = 2(1 - Weight_1)$$

The objective I_1 increases with x_1 whereas I_2 decreases. Thus, these are conflicting in nature.

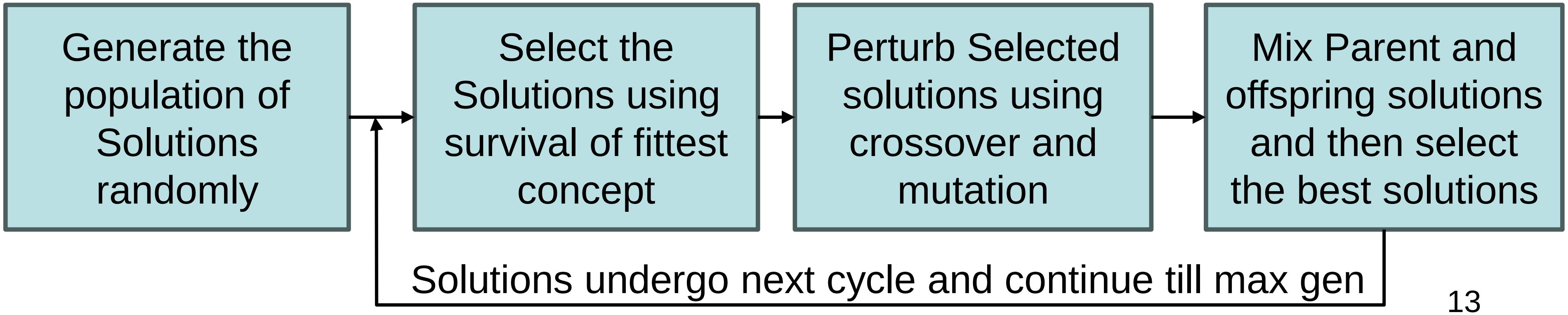
Genetic Algorithm

1) Based on Concept of Evolution and Genetics.
2) Population of Solutions is Generated initially. Solutions are Perturbed using genetic operations to generate Offspring Solutions. Only fittest solutions are allowed to survive for the next generation. Procedure is repeated over the generations to obtain the optimal solutions.
3) It is non-derivative based approximate algorithm and converges asymptotically to the global optimal solution.
4) It can handle multi-objective, non-linear and multi-solution formulation easily.

Childrens enrolled in Class 1

Childrens gain knowledge from teachers, peers, books

Better students pass examination and go to next class



Generate the population of Solutions randomly

Select the Solutions using survival of fittest concept

Perturb Selected solutions using crossover and mutation

Mix Parent and offspring solutions and then select the best solutions

Solutions undergo next cycle and continue till max gen

Case Study

Background: Selecting the best team based on past performance and consistency is an extremely important problem. This is primarily due to the advent of several online gaming platforms such as Dream 11, Fantasy League, etc.

In this case study, a dataset comprising the past performance of 15 Indian and 15 English players for the past 10 matches is given along with the price in crore.

Name	Role	Country	Score1	Score2	Score3	Score4	Score5	Score6	Score7	Score8	Score9	Score10	Price
Virat Kohli	BAT	IND	145	128	41	55	85	56	3	122	4	54	17
Rohit Sharma	BAT	IND	46	48	86	131	0	81	6	53	56	74	15
Shubman Gill	BAT	IND	26	53	16	104	74	27	121	19	58	67	8
Ravindra Jadeja	AR	IND	39	50	50	16	75	8	75	25	25	50	16
Jasprit Bumrah	BOWL	IND	25	50	50	100	50	75	25	25	50	25	12
JB Bairstow	WKT	ENG	30	10	2	52	33	13	0	6	28	63	6

Case Study

Problem:

For the given data on performance scores of 30 players, select a team of 11 players with a budget of 100 Crore. Also, satisfy the constraints such as batsman and bowlers should be between 3 – 6, and all-rounder and wicket keeper should be between 1 – 4. Further, no more than 7 players should be from one country.

1) Perform multi-objective optimization to maximize performance score (Give weightage 2 for caption, 1.5 for vice-captain, and 1 for normal player) and minimize the risk (minimize the standard deviation) using a weighted sum method.
2) Analyze the Pareto optimal results obtained.
3) Select the suitable operating point on the Pareto optimal front and obtain the corresponding team that is consistent yet can give reasonable performance.

AIM: To understand the concept of multi-objective optimization and genetic algorithm.

Applications:

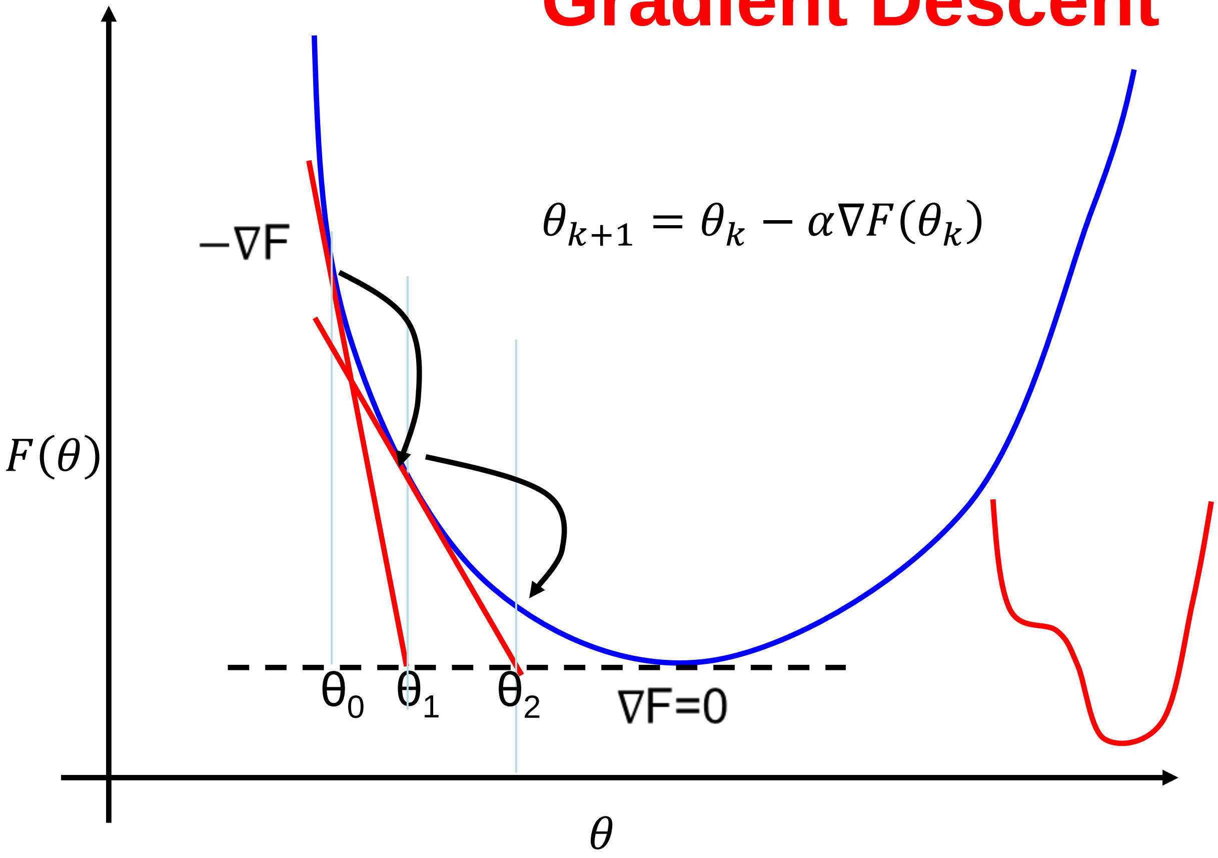
Selecting the best team from past performance is extremely important for making a wise decision.

Optimizers Used in Machine Learning

1) Gradient Descent
2) Nesterov Accelerated Gradient (NAG)
3) Adaptive Gradient Algorithm (AdaGrad)
4) Root Mean Square Propagation (RMS-Prop)
5) Adaptive Moment Estimation (Adam)

All Machine learning optimizers used are the extension of Gradient Descent.

Gradient Descent



$$\theta_{k+1} = \theta_k - \alpha \nabla F(\theta_k)$$

Concept of Momentum

$$v_k = \gamma v_{k-1} + \alpha \nabla F(\theta_k)$$

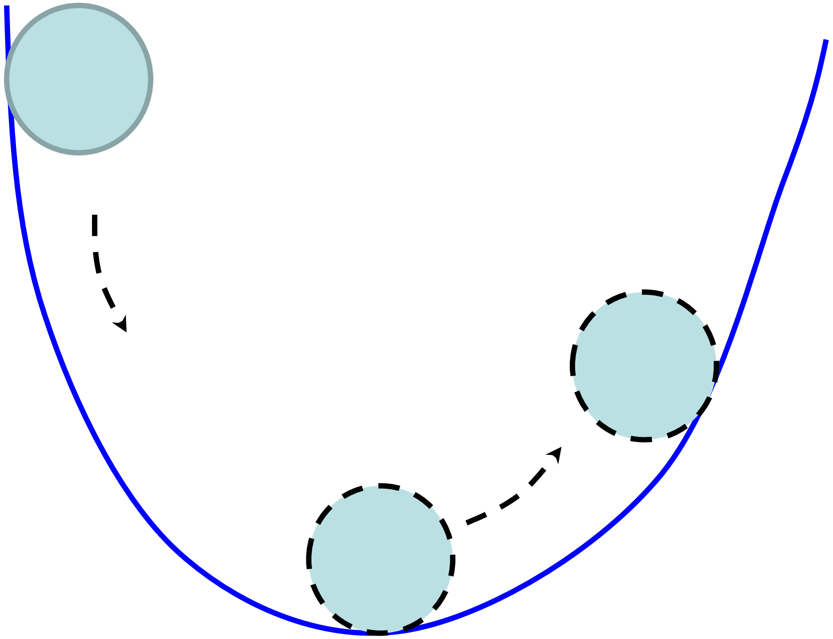
$$\theta_{k+1} = \theta_k - v_k$$

1) A typical value of γ used is 0.9.
2) Concept is like rolling a ball from a hill to valley while finding the minima.

1) Vanilla/ Batch Gradient Descent: Calculates the averaged gradient with all training samples.
2) Stochastic Gradient Descent: Calculates the gradient with one training sample. Response is memory efficient but unstable (response fluctuates) and may shoot even after getting global optimum. Concept of momentum reduces the fluctuations.
3) Mini Batch Gradient Descent: The data is divided into different mini batches. Gradient is calculated as average value over all samples present in one mini batch.

Nesterov Accelerated Gradient (NAG)

1) Requires a more efficient way than just rolling a ball from hill to valley while finding the minimum. When we reach the minimum there is a need to slow down otherwise the ball will not stop at flat surface at the minimum and will continue to move up.
2) NAG provides an ability of slowing down when algorithm reaches close to minimum.
3) It works better than the conventional momentum algorithm.



Gradient is calculated with respect to estimated future position

$$\theta_{k+1,estimated} = \theta_k - \gamma v_{k-1}$$

$$v_k = \gamma v_{k-1} + \alpha \nabla F\left(\theta_{k+1,estimated}\right)$$

$$\theta_{k+1} = \theta_k - v_k$$

Adaptive Gradient Algorithm (AdaGrad)

1) It adapts different learning rates for different parameters. The parameters associated with frequently occurring features are adapted with smaller learning rates whereas the parameters associated with less frequent features are updated with larger learning rates.
2) The learning rate decreases with increase in number of steps.
3) It is well suited for the cases in which data is sparse.

$$g_{0,i} = 0$$

$$g_{k,i} = g_{k-1,i} + \left[\nabla F\left(\theta_{k,i}\right)\right]^2$$

$$\theta_{k+1,i} = \theta_{k,i} - \frac{\alpha \nabla F\left(\theta_{k,i}\right)}{\sqrt{g_{k,i} + \epsilon}}$$

4) It does not require the tuning of learning rate. α is kept at 0.01.
5) Main disadvantage is the accumulation of positive square gradient term in each step makes $g_{k,i}$ term very large which makes the learning rate very small over the iterations.

AdaGrad is extended to AdaDelta and RMS-Prop in which the aggressively decreasing learning rate is reduced.

$$g_{0,i} = 0$$

γ used is 0.9.

$$g_{k,i} = \gamma g_{k-1,i} + (1-\gamma)\left[\nabla F\left(\theta_{k,i}\right)\right]^2$$

AdaDelta:

$$\theta_{k+1,i} = \theta_{k,i} - \frac{\sqrt{g_{k-1,i} + \epsilon}}{\sqrt{g_{k,i} + \epsilon}} \nabla F\left(\theta_{k,i}\right)$$

RMS-Prop:

$$\theta_{k+1,i} = \theta_{k,i} - \frac{\alpha \nabla F\left(\theta_{k,i}\right)}{\sqrt{g_{k,i} + \epsilon}}$$

Adaptive Moment Estimation (Adam)

1) It adapts different learning rates for different parameters. In addition to storing of square of gradient for calculating the learning rate (similar to RMS-Prop), it also adapts the gradient with the concept of momentum.

$$m_{0,i} = 0, g_{0,i} = 0$$

$$m_{k,i} = \beta_1 m_{k-1,i} + (1-\beta_1)\nabla F\left(\theta_{k,i}\right)$$

$$g_{k,i} = \beta_2 g_{k-1,i} + (1-\beta_2)\left[\nabla F\left(\theta_{k,i}\right)\right]^2$$

$$M_{k,i} = \frac{m_{k,i}}{(1-\beta_1)}; G_{k,i} = \frac{g_{k,i}}{(1-\beta_2)}$$

$$\theta_{k+1,i} = \theta_{k,i} - \frac{\alpha}{\sqrt{G_{k,i} + \epsilon}} M_{k,i}$$

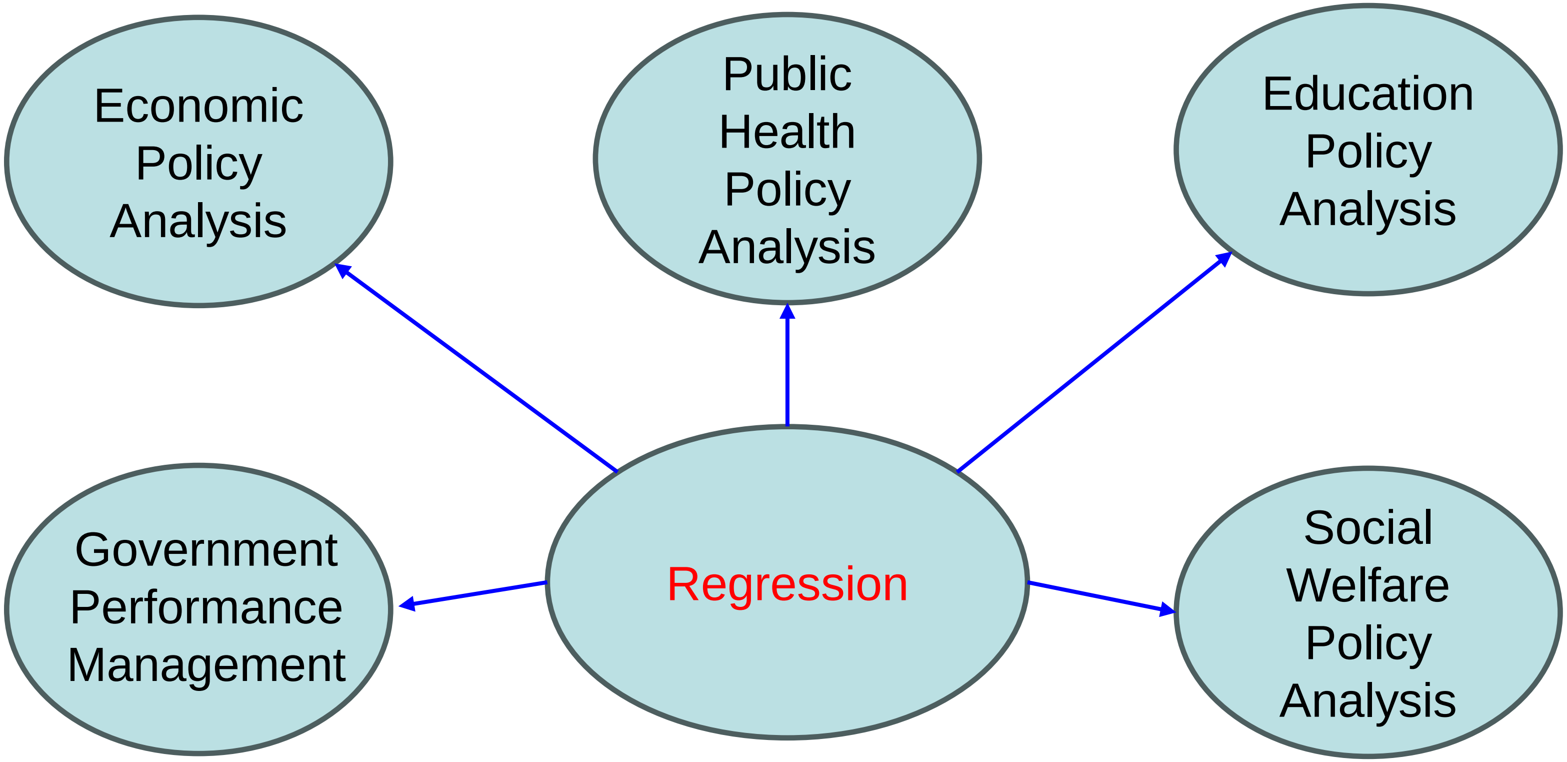
2) Typical values used are β_1= 0.9, β_2 = 0.999, and ∈ = 10^{-8}.
3) Selection of the optimizer depends on the data. Typically for sparse data, adaptive learning methods are best. Further, if learning rate is diminishing fast, then the methods such as RMS-Prop, AdaDelta and Adam are good. Overall, Adam is best among all.

Application of Regression

GDP growth, unemployment rates, and inflation

Impact of access to healthcare, public health campaigns, and environmental factors

Resource allocation and curriculum development



Economic Policy Analysis
Public Health Policy Analysis
Education Policy Analysis
Government Performance Management
Regression
Social Welfare Policy Analysis

Relationship between inputs (such as budget allocations and staffing levels) and outputs

Social programs and interventions aimed at reducing poverty, inequality, and social exclusion

Case Study on Regression

Background: The government of India and different State Governments have rolled out several health insurance schemes for the citizens. Some of these are Ayushman Bharat, and Pradhan Mantri Jeevan Jyoti Bima Yojana, etc. These schemes aim to provide affordable healthcare coverage to citizens across the country. As policymakers gear up to roll out new schemes or enhance existing ones, they can rely on data-driven insights to make informed decisions.

In this case study, a dataset involving 1338 data points with input features (age, sex, BMI, children, smoker, region) and output feature (charges) is given. Complete data will be given in a CSV file.

Age	Sex	BMI	Children	Smoker	Region	Charges
19	female	27.9	0	yes	southwest	16420
18	male	33.77	1	no	southeast	15288.5
28	male	33	3	no	southeast	27750
33	male	22.705	0	no	northwest	32225
32	male	28.88	0	no	northwest	32044
31	female	25.74	0	no	southeast	30312
46	female	33.44	1	no	southeast	60072
37	female	27.74	3	no	northwest	42112
37	male	29.83	2	no	northeast	41716.5
60	female	25.84	0	no	northwest	96292

Case Study on Regression

Problem:

In this scenario, you're presented with a dataset containing crucial information on individuals' health and insurance charges. With these, develop the predictive model using regression as given below:

1) Visualize the impact of different features on output.
2) Clean the data by removing outliers.
3) Use the cleaned data to fit the linear regression model.
4) Use the cleaned data to fit the polynomial regression model.
5) Evaluate the regularized regression on the cleaned data.
6) Evaluate the importance of different features in predicting the insurance charges.

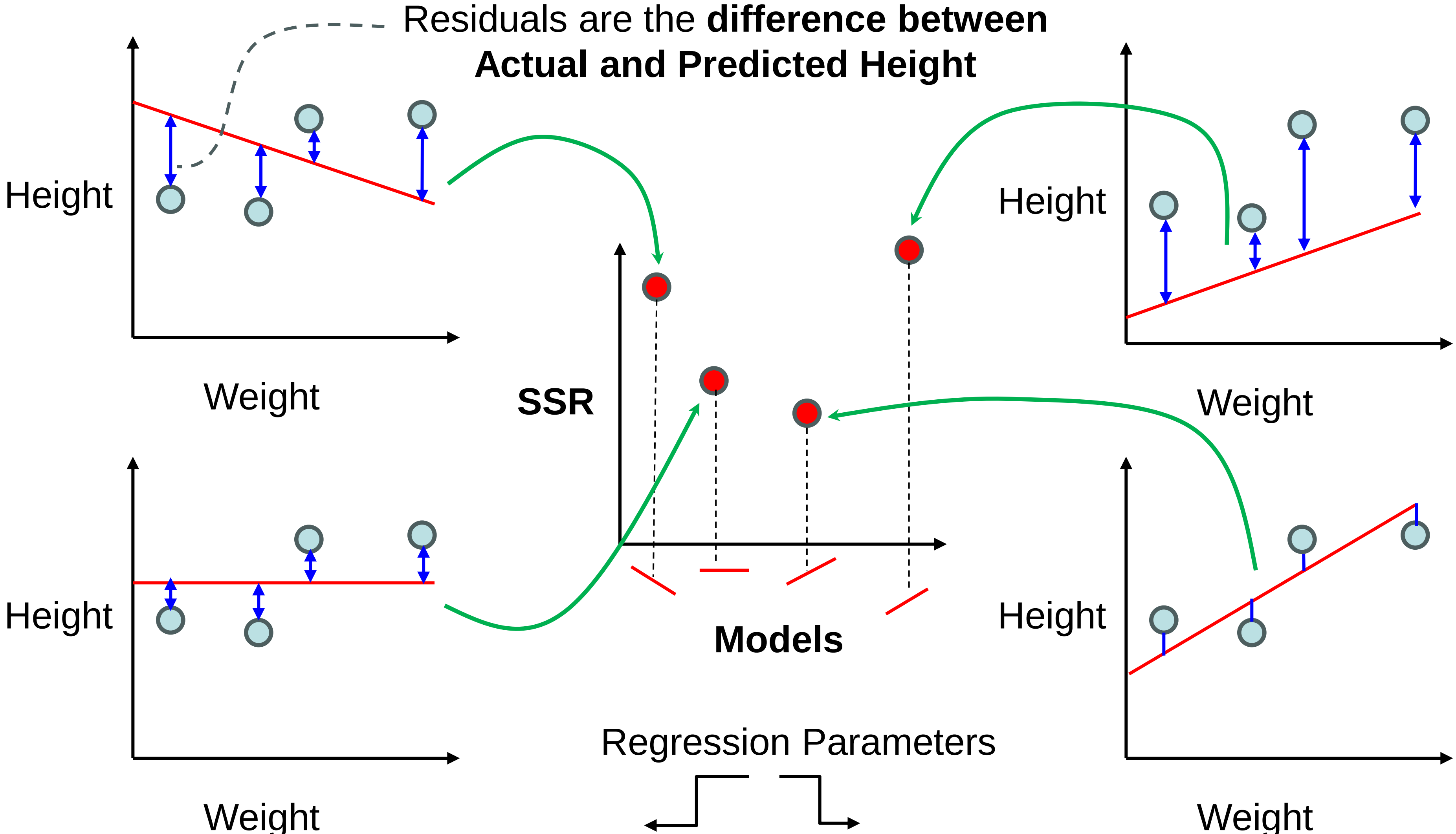
AIM: To understand the concept of linear, polynomial, and regularized regression.

Applications:

Developing the regression model and predicting the insurance charges is an important problem for decision-makers, particularly while rolling out the new scheme.

Linear Regression

Linear Regression fits the linear line to the given data that minimizes the **sum of the square residuals (SSR)**.

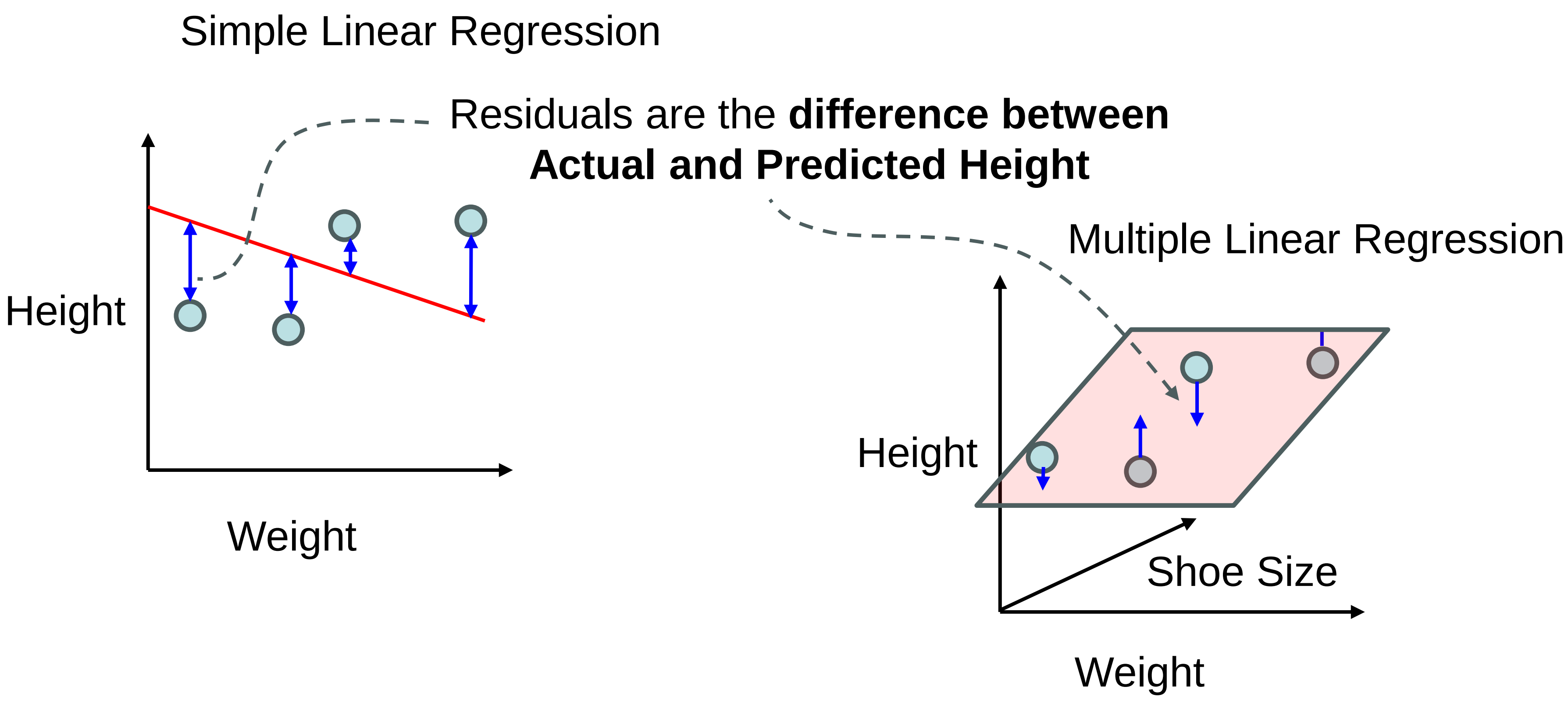


Residuals are the **difference between Actual and Predicted Height**

Height = Intercept (θ_0) + Slope (θ_1) × Weight

Multiple Linear Regression

Multiple Linear Regression fits the linear plane to the given data that minimizes the **sum of the square residuals (SSR)**.



Simple Linear Regression

Residuals are the **difference between Actual and Predicted Height**

Multiple Linear Regression

Height = Intercept (θ_0) **+** Slope1 (θ_1) **×** **Weight +** Slope2 (θ_2) **× Shoe Size**

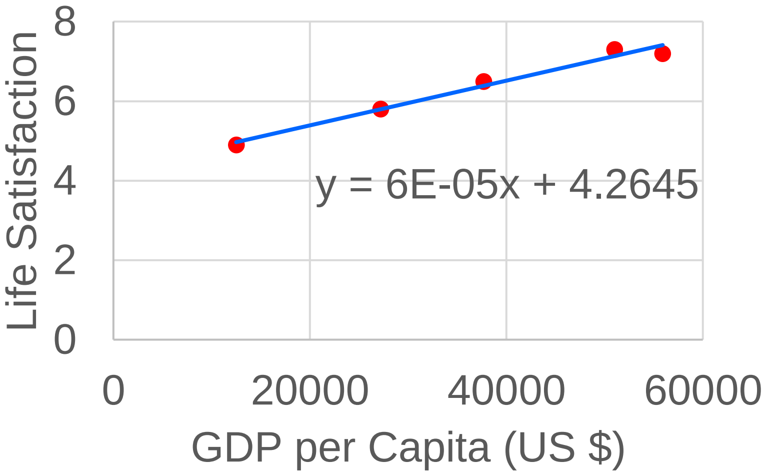
$$R^2 = \frac{SSR(Mean\ Height) - SSR(Fitted\ Line)}{SSR(Mean\ Height)}$$

Linear Regression

Country	Hungary	Korea	France	Australia	United States
Life satisfaction (LS)	4.9	5.8	6.5	7.3	7.2
GDP per capita (US $)	12500	27200	37700	51000	55900

$$LS = \theta_0 + \theta_1(GDP)$$

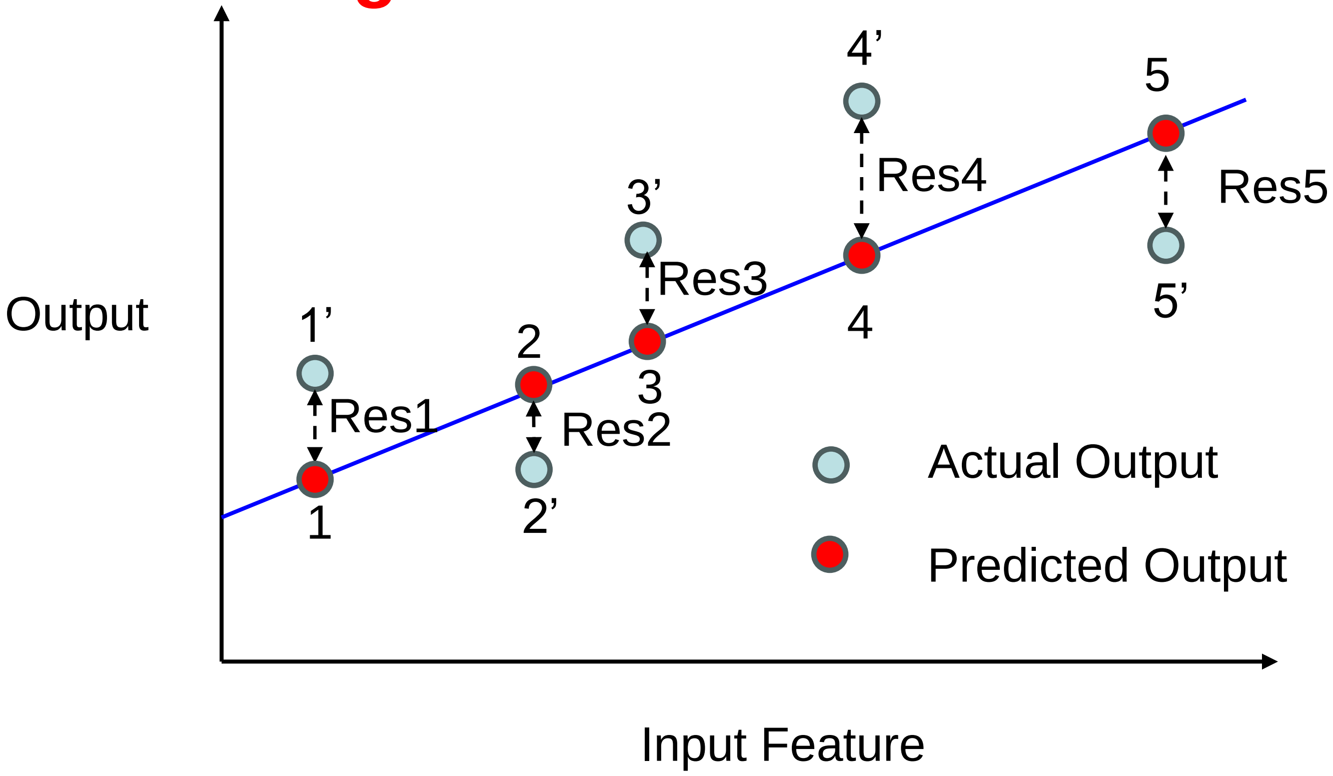
Linear regression involves a linear model in which prediction is made by simply computing a weighted sum of the input features, plus a bias term (i.e., intercept, θ_0).



Predicted output (y') = Bias (θ_0) + Weight 1 (θ_1) × Feature 1 (x_1) + Weight 2 (θ_2) × Feature 2 (x_2) + ….+ Weight n (θ_n) × Feature n (x_n)

More concisely, one can write: $y' = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$ Where, x_0 = 1.

Linear Regression with 5 data Points



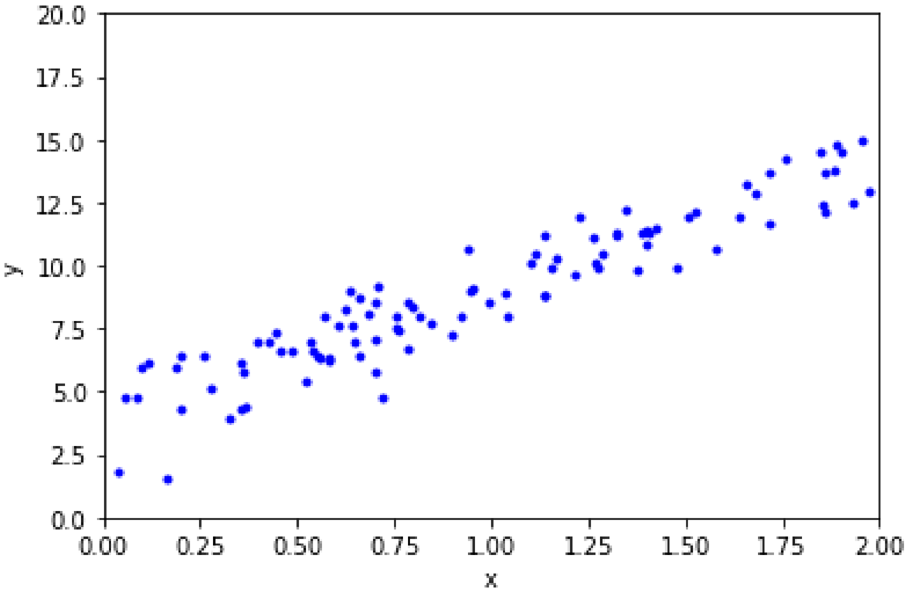
Min MSE = minimize (Res1^2 + Res2^2 + Res3^2 + Res4^2 + Res5^2)/5

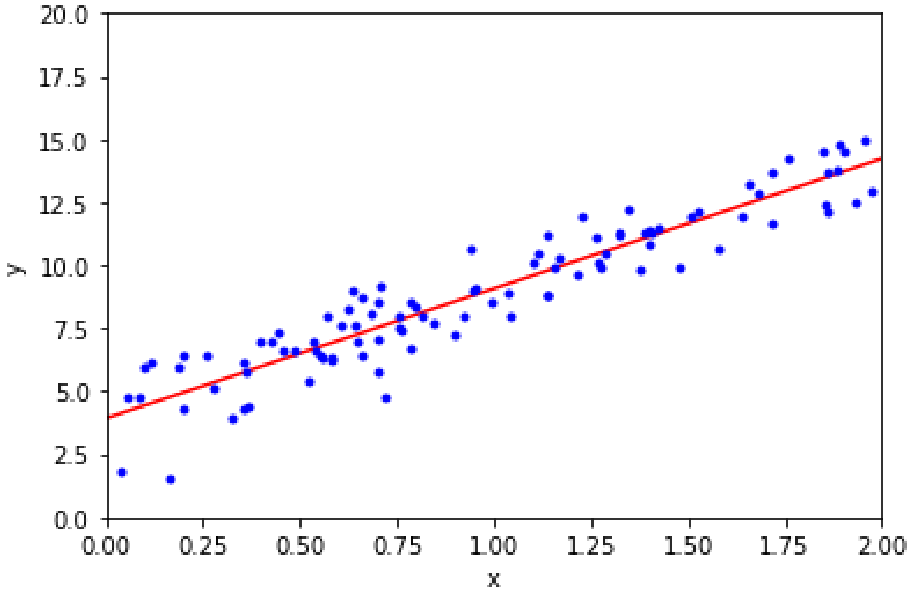
Closed form solution: $\boldsymbol{\theta}^{opt} = (X^T X)^{-1}(X^T y)$

Best values of weights θ_1, θ_1, …, θ_n are obtained by closed form solution.

Linear Regression

1) To illustrate the concept, consider a simple example of fitting the data as y = $\theta_0 + \theta_1 x$
2) Artificially, the data of 100 size is generated as x = 2× rand(); y = 4 + 5x + rand()





Best fit is obtained as: y = 3.9229 + 5.1540x

Interpretation of Linear Regression

1) Let us assume that the outputs and the inputs are related via

$$y^{(i)} = \boldsymbol{\theta}^T \boldsymbol{x}^{(i)} + \in^{(i)}$$

2) $\in^{(i)}$ is an error term that captures either unmodeled effects, or random noise and are distributed IID (independently and identically distributed) according to a Gaussian distribution.

So, $\in^{(i)} \sim N(0, \sigma^2)$, that is, $$p\left(\in^{(i)}\right) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{1}{2} \frac{\left(\in^{(i)}\right)^2}{\sigma^2} \right)$$

Hence, $$p\left(y^{(i)} \middle| \boldsymbol{x}^{(i)}; \boldsymbol{\theta}\right) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{1}{2} \frac{\left(y^{(i)} - \boldsymbol{\theta}^T \boldsymbol{x}^{(i)}\right)^2}{\sigma^2} \right)$$

3) Given X (which contains all the $\mathbf{x}^{(i)}$'s) and θ, what is the distribution of the $y^{(i)}$? The probability of this is: p(Y |X; **θ**)

Interpretation of Linear Regression

1) Likelihood function: explicit representation of this as a function of **θ**:

$$L(\boldsymbol{\theta}) = L(\boldsymbol{\theta}, X, Y) = p(Y|X;\boldsymbol{\theta}) = \prod_{i=1}^{m} p\left(y^{(i)}\middle|\boldsymbol{x}^{(i)};\boldsymbol{\theta}\right)$$

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{1}{2}\frac{\left(y^{(i)} - \boldsymbol{\theta}^T \boldsymbol{x}^{(i)}\right)^2}{\sigma^2}\right)$$

2) Maximum likelihood estimate of **θ: ** Maximize L(**θ**)
3) Loglikelihood l(**θ**) = ln L(**θ**) - for simplicity and tractability

MLE: $$I(\boldsymbol{\theta}) = \ln \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{1}{2}\frac{\left(y^{(i)} - \boldsymbol{\theta}^T \boldsymbol{x}^{(i)}\right)^2}{\sigma^2}\right)$$

$$= m \ln \frac{1}{\sqrt{2\pi\sigma}} - \frac{1}{2\sigma^2}\sum_{i=1}^{m}\left(y^{(i)} - \boldsymbol{\theta}^T \boldsymbol{x}^{(i)}\right)^2$$

4) Maximizing likelihood is equivalent to minimizing the square of error. Least square regression is special case of MLE.

Locally Weighted Linear Regression

1) Objective is to fit the model preferentially at local region. For instance, if a large range of data is given, and one want to fit the linear model preferentially around a selected data point.:

$$\min MSE\left(\boldsymbol{\theta}\right)=\frac{1}{m}\sum_{i=1}^{m}w^{(i)}\left(\boldsymbol{\theta}^{T}\boldsymbol{x}^{(i)}-y^{(i)}\right)^{2}$$

Where, $w^{(i)}$ are the weights corresponding to the data points.

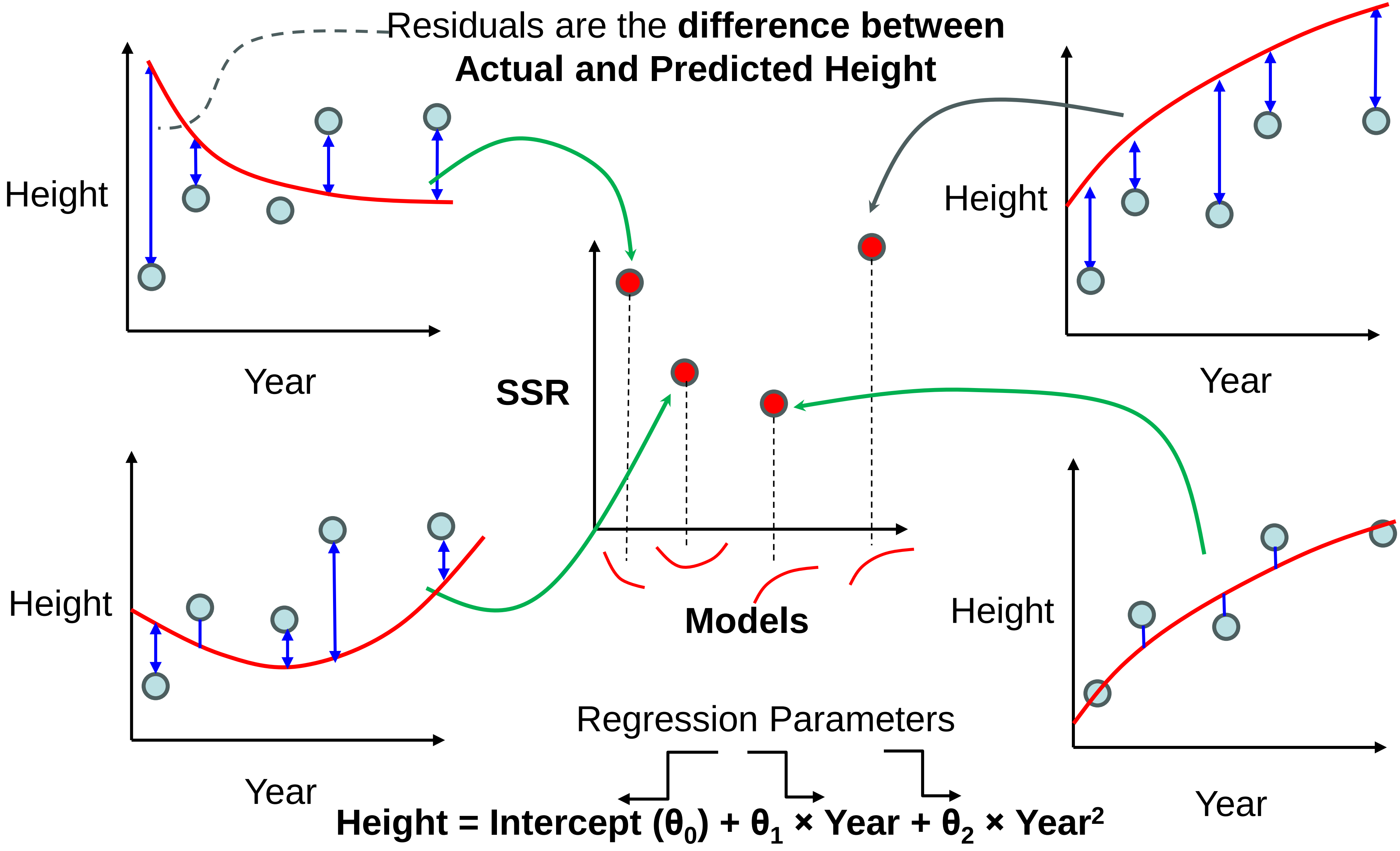
2) If $w^{(i)}$ is large, the penalization of $(\boldsymbol{\theta}^T\mathbf{x}^{(i)} – y^{(i)})^2$ is large and when $w^{(i)}$ is small the penalization of $(\boldsymbol{\theta}^T\mathbf{x}^{(i)} – y^{(i)})^2$ is small.
3) The standard choice of $w^{(i)}$ at a particular query point is:

$$w^{(i)}=\exp\left(-\frac{\left(\boldsymbol{x}^{(i)}-\boldsymbol{x}\right)^{2}}{2\tau^{2}}\right)$$

4) For small $|\mathbf{x}^{(i)} - \mathbf{x}|$, $w^{(i)} \approx 1$. For large $|\mathbf{x}^{(i)} - \mathbf{x}|$, $w^{(i)} \approx 0$.
5) The bandwidth parameter τ which decides how the weight of a training example falls off with distance of its $\mathbf{x}^{(i)}$ from the query point $\mathbf{x}$.

Polynomial Regression

Polynomial Regression fits the Curve to the given data that minimizes the **sum of the square residuals (SSR)**.

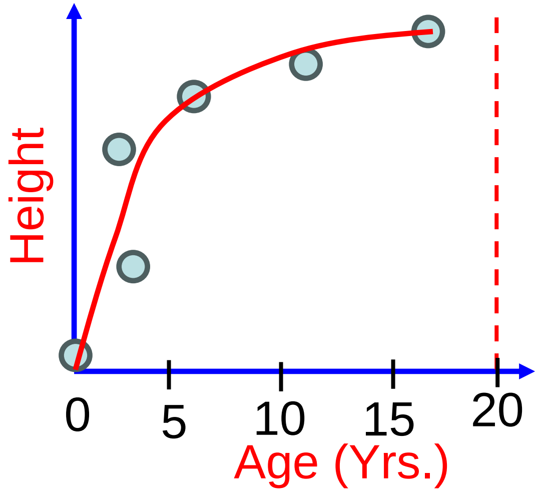


Residuals are the **difference between Actual and Predicted Height**

Regression Parameters

Height = Intercept (θ_0) + θ_1 × Year + θ_2 × Year^2

Polynomial Regression



1) Interesting to note that the linear regression can be used for representing such complex data by representing the ‘feature with powers’ by new features and then taking a linear combinations of these.

$$\text{Height} = \text{Intercept} + \theta_1 \times \text{Age} + \theta_2 \times \text{Age}^2$$

The predicted output is represented in terms of power of feature i.e., height is represented as polynomial of Age

To convert the problem to linear regression: Age = Feature 1; Age^2 = Feature 2

$$\text{Height} = \text{Intercept } (\theta_0) + \theta_1 \times \text{Feature 1} + \theta_2 \times \text{Feature 2}$$ (Linear Equation)

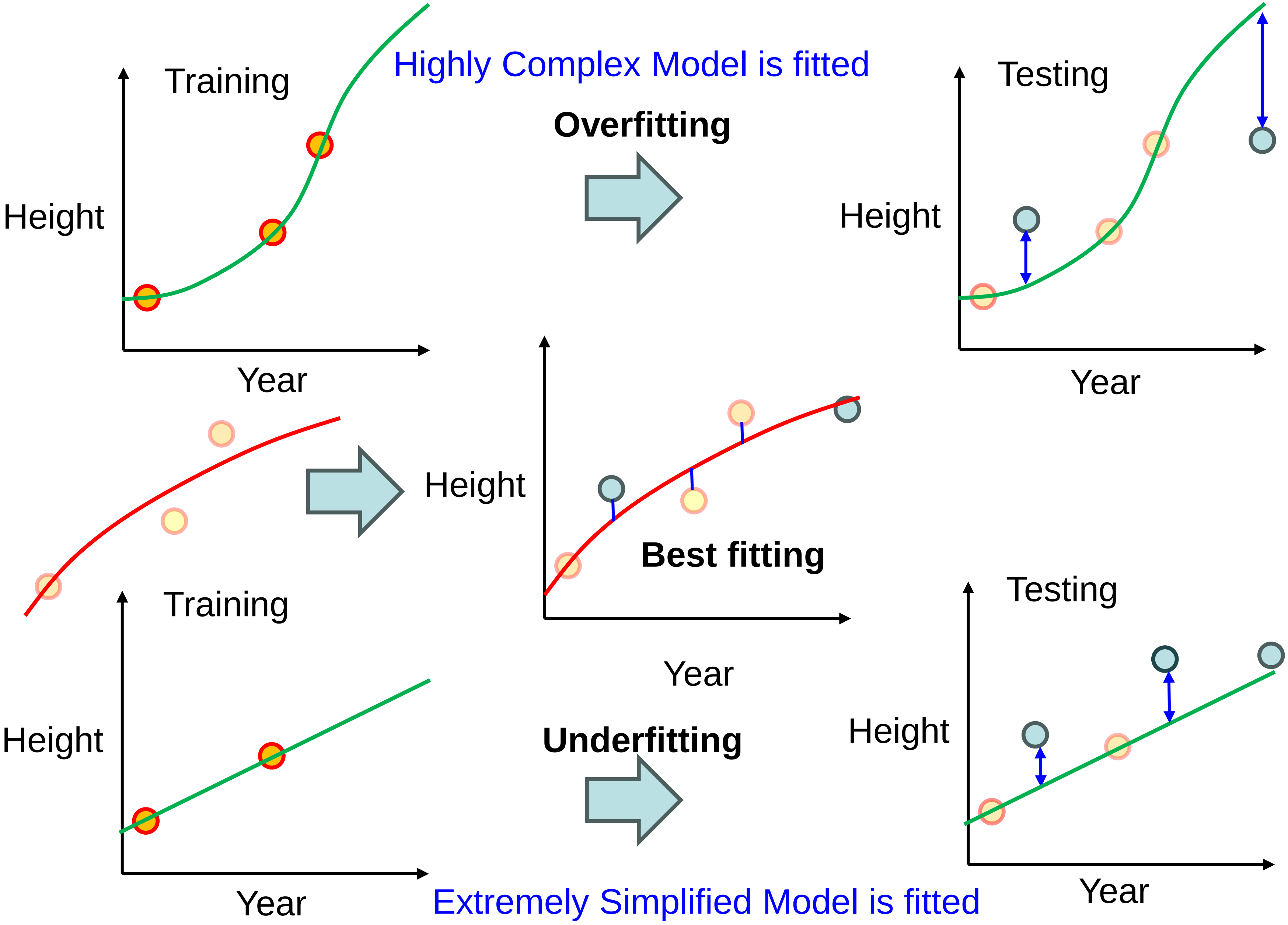
2) A polynomial of any degree for given number of features can be given as input to fit the data. For instance, if degree = 2 and features = 2 (Age, and Shoe Size) then the output will be

Features are correlated

$$\text{Height} = \theta_0 + \theta_1 \times \text{Age} + \theta_2 \times \text{Shoe Size} + \theta_3 \times (\text{Age} \times \text{Shoe Size}) + \theta_4 \times \text{Age}^2 + \theta_5 \times \text{Shoe Size}^2$$

3) Challenge is to obtain correct degree of the polynomial which do not cause underfitting or overfitting.

Under Fitting & Overfitting



Highly Complex Model is fitted

Overfitting

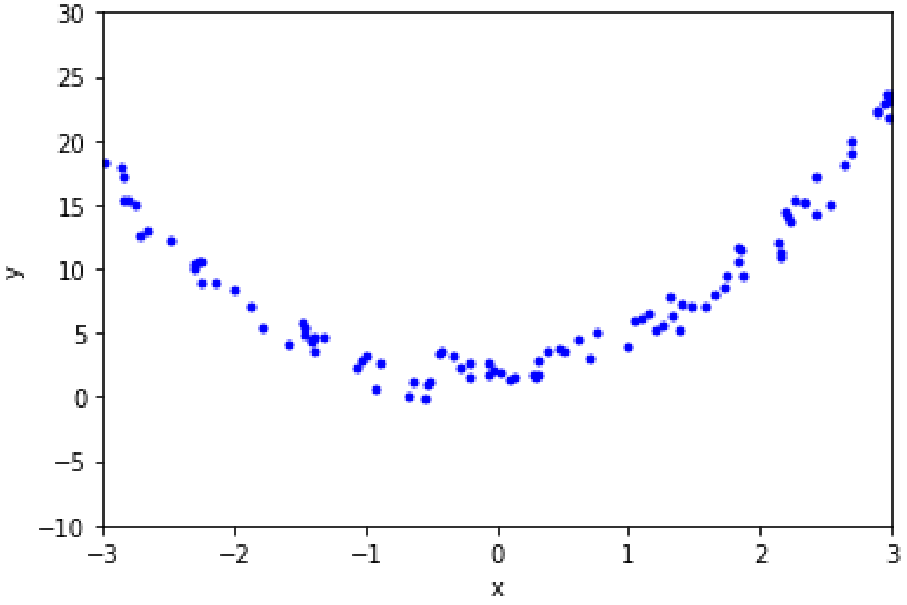
Best fitting

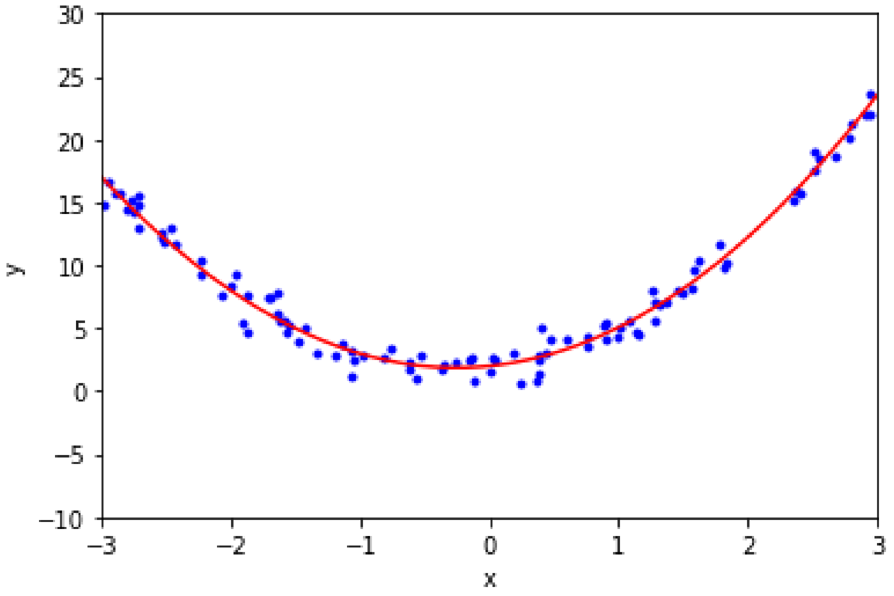
Underfitting

Extremely Simplified Model is fitted

Polynomial Regression

1) For illustration, consider fitting $y' = \theta_0+\theta_1x+\theta_2x^2$ to the data.
2) Artificially, the data of 100 size is generated as $x = 6\times \text{rand}() - 3$; $y = 2 + x + 2x^2 + \text{rand}()$

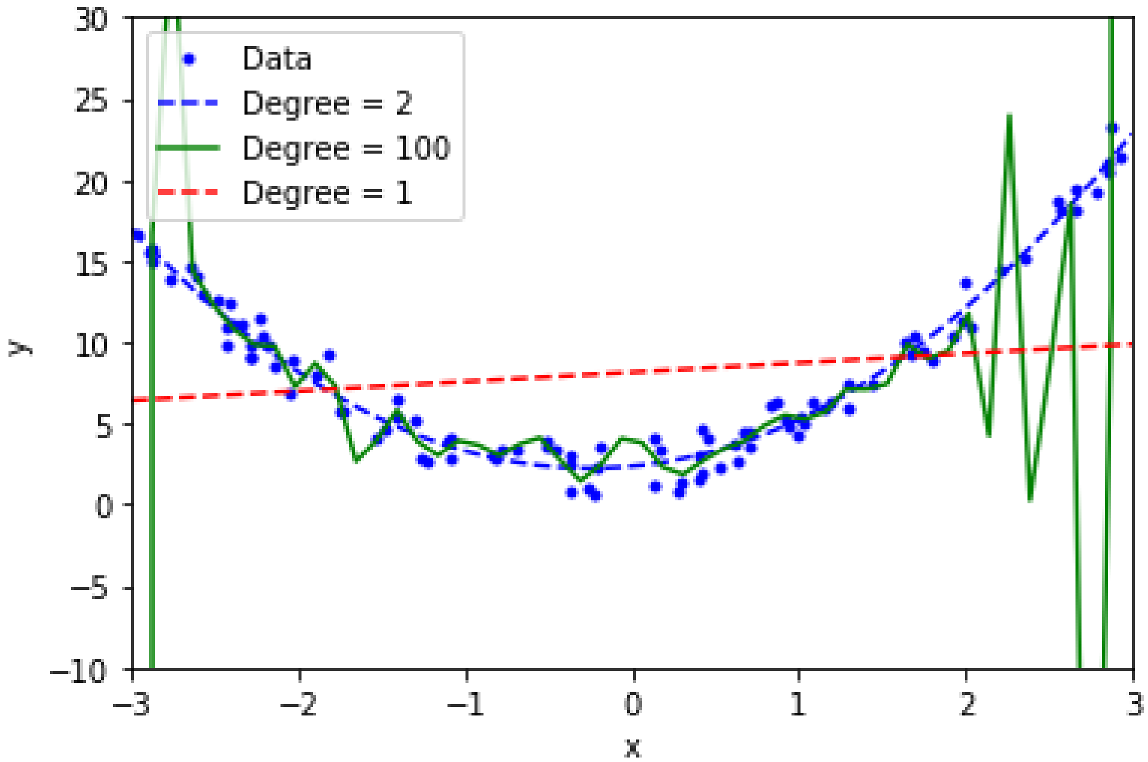




Best fit is obtained as: $y = 1.775 + 0.9763x+2.02x^2$

Polynomial Regression

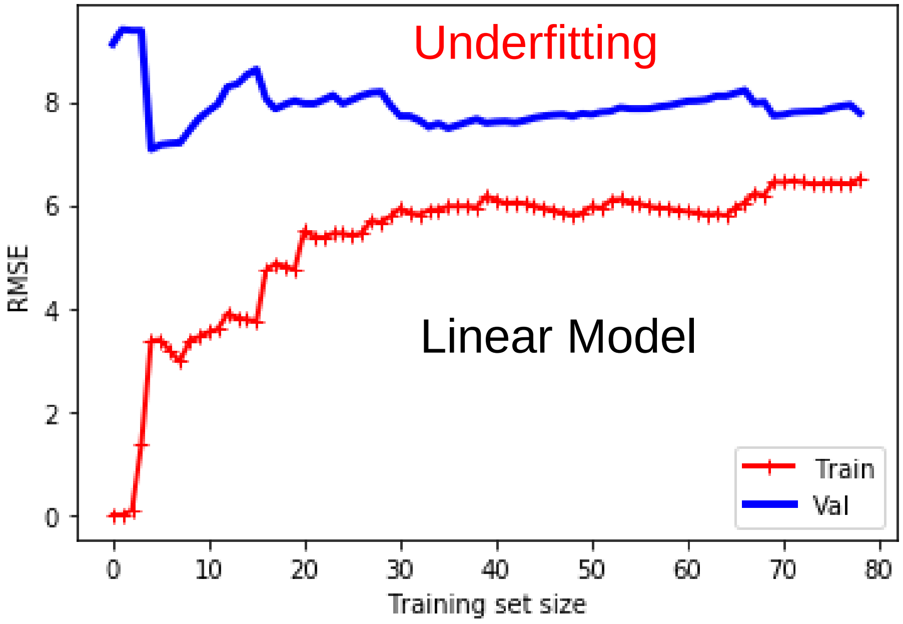
1) With increase in degree, the polynomial curve tries to fit a every data points resulting into wiggling of the model.

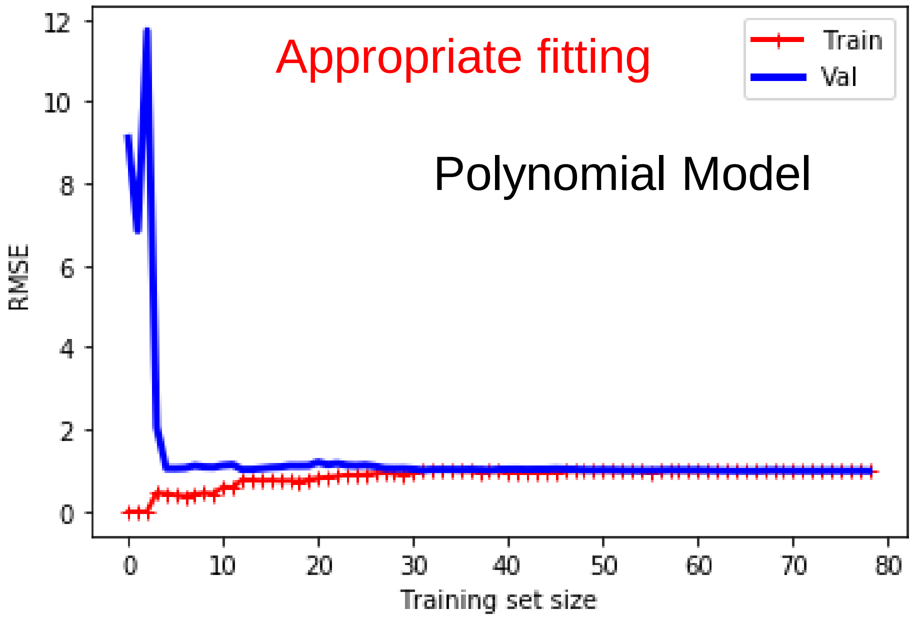


1) Most appropriate fitting occurs with degree = 2 as we know that the data is generated with degree = 2. However, in real life case this information is not known, and one must evaluate the most appropriate value of degree which do not cause overfitting or underfitting.

Polynomial Regression

1) To evaluate whether the model is overfitting or under-fitting is extremely important question.
2) One can evaluate whether the model is overfitting or underfitting from learning curves.





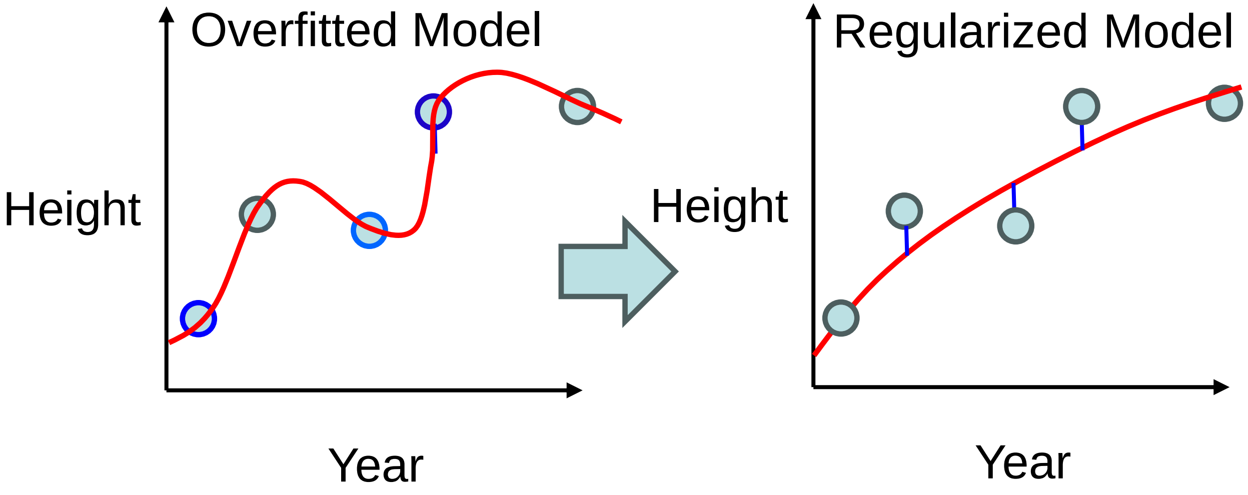
1) When settling error for both training and validation is high then it shows underfitting.
2) When the gap between training and validation curves is large then it shows overfitting.

Regularization

1) Regularization of the model is required to eliminate the overfitting.
2) When there are many correlated variables in a linear regression model, their coefficients can become poorly determined and exhibit high variance.
3) A wildly large positive coefficient on one variable can be canceled by a similarly large negative coefficient on its correlated variable. By imposing a size constraint on the coefficients, this problem can be alleviated.
4) Three popular ways of regularizing the models: ridge regression, LASSO (least absolute shrinkage and selection operator) regression and elastic net.

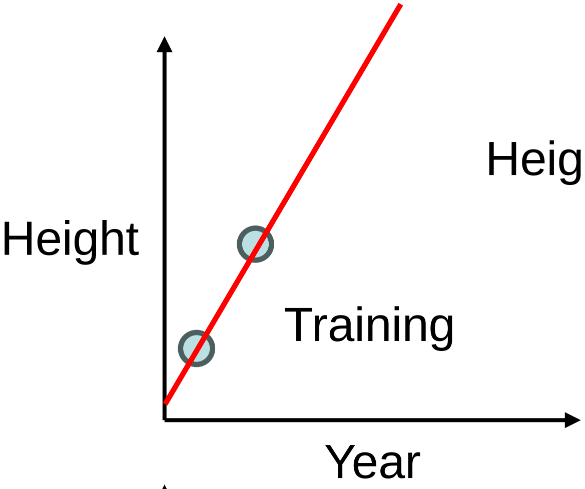
We start with overfitted model and remove the overfitting automatically using regularization.

Regularization = Reducing the values of weight factors (θ_1, θ_1, …, θ_n) or sometimes even eliminating some of the features.

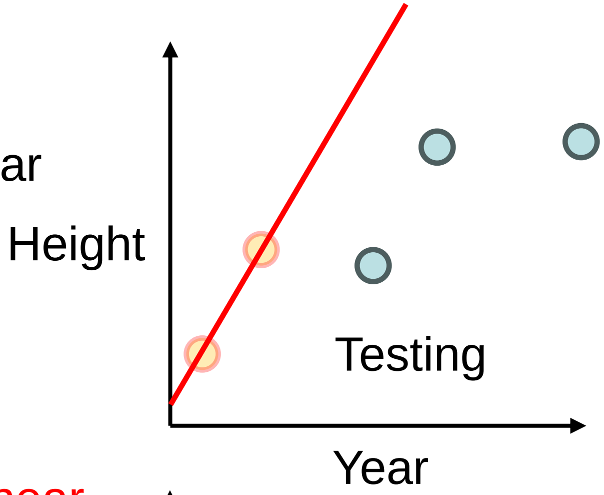


Regularization

Height = Intercept + Weight × Year

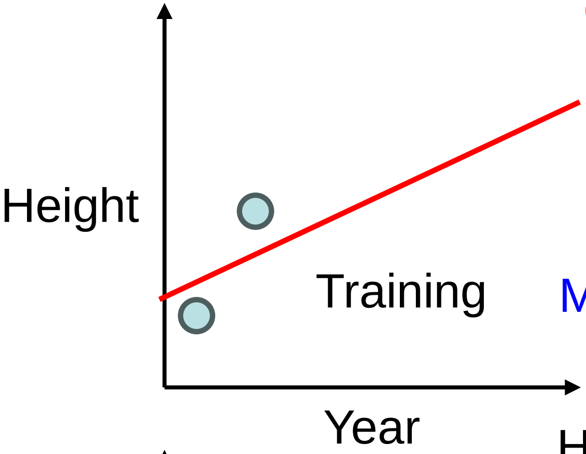


Minimize SSR (Sum of Square Residuals)

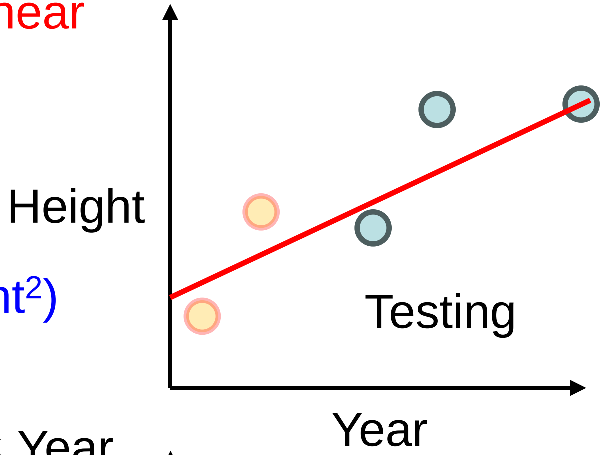


Concept Explained using Linear Regression

Ridge

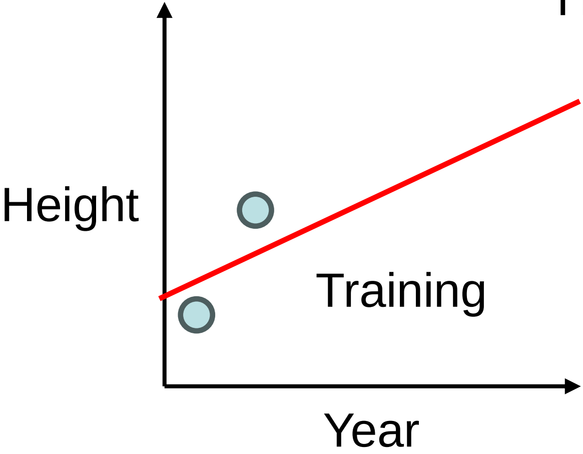


Minimize (SSR + α/2 × Weight^2)

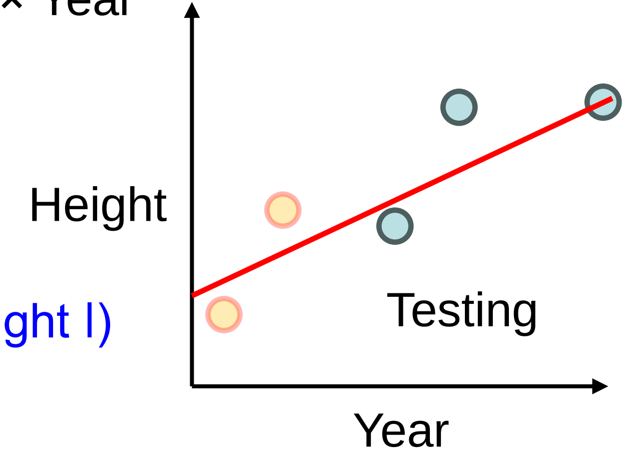


Height = Intercept + Weight × Year

LASSO



Minimize (SSR + α × | Weight |)



Ridge Regression

1) It represents a regularized version of linear regression.
2) It forces not only fitting of the data but also keep the weights as small as possible.
3) It is to be noted that the regularization term is added (half of the square of the L_2 norm of the weight vector) only for the training of the model. Once the trained model is obtained, it is fit to the validation data without the regularization terms to evaluate its performance.

$$\min J(\boldsymbol{\theta}) = MSE(\boldsymbol{\theta}) + \frac{\alpha}{2}(\theta_1^2 + \theta_2^2 + \cdots + \theta_n^2)$$

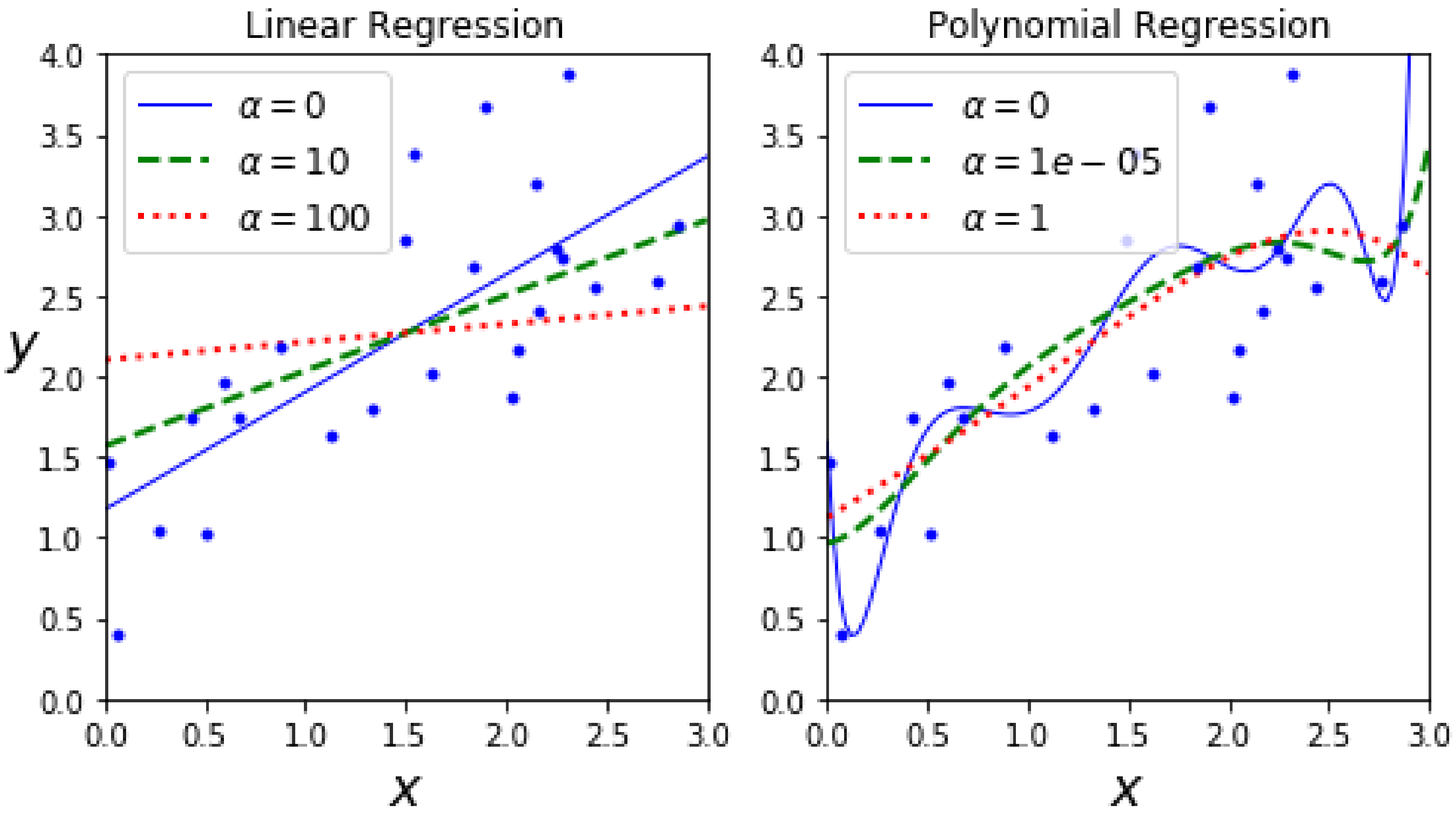
Regularization term

4) Note that the bias term θ_0 is not regularized.
5) The closed form solution is given by: $\boldsymbol{\theta}^{opt} = (X^T X + \alpha A)^{-1}(X^T y)$

Here, $\boldsymbol{\theta}^{opt}$ are the best values of **θ** that minimizes the objective function J. Also, **A** is (n+1) × (n+1) identity matrix except with 0 term in the top left for bias term.

4) The solution adds a positive constant to the diagonal of $X^T X$ before inversion. This makes the problem nonsingular, even if $X^T X$ is not of full rank, and was the main motivation for ridge regression.

Ridge Regression



Increase in α values flattens the curves and increases the bias

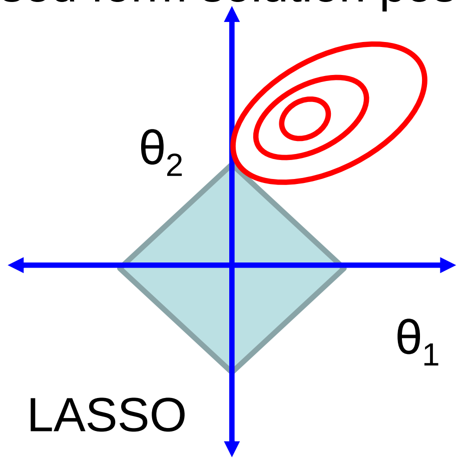
LASSO Regression

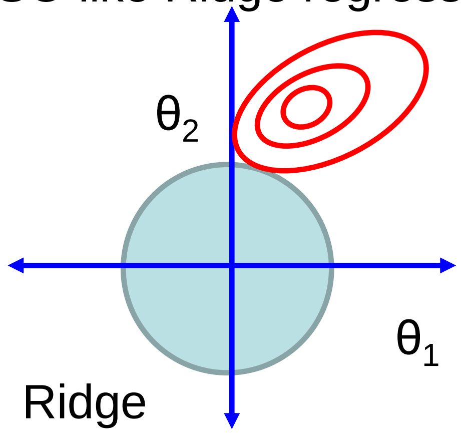
1) It represents another regularized version of linear regression similar to ridge regression.

$$\min J(\boldsymbol{\theta}) = MSE(\boldsymbol{\theta}) + \alpha(|\theta_1| + |\theta_2| + \cdots + |\theta_n|)$$

Regularization term

2) Note that the bias term θ_0 is not regularized.
3) L_1 norm of the weight vector is added for regularizing the MSE. This L_1 penalty shrinkage is much more than that obtained using half of the square of L_2.
4) This latter constraint makes the solution non-linear in $y^{(i)}$ and there is no closed form solution possible for LASSO like Ridge regression.





LASSO Regression

1) For the simple case having two parameters θ$_1$ and θ$_2$, the residual sum of squares has elliptical contours, centered at the full least squares estimate.
2) Constraint for ridge regression is disk:

$$\theta_1^2 + \theta_2^2 \leq User\ Defined\ Threshold$$

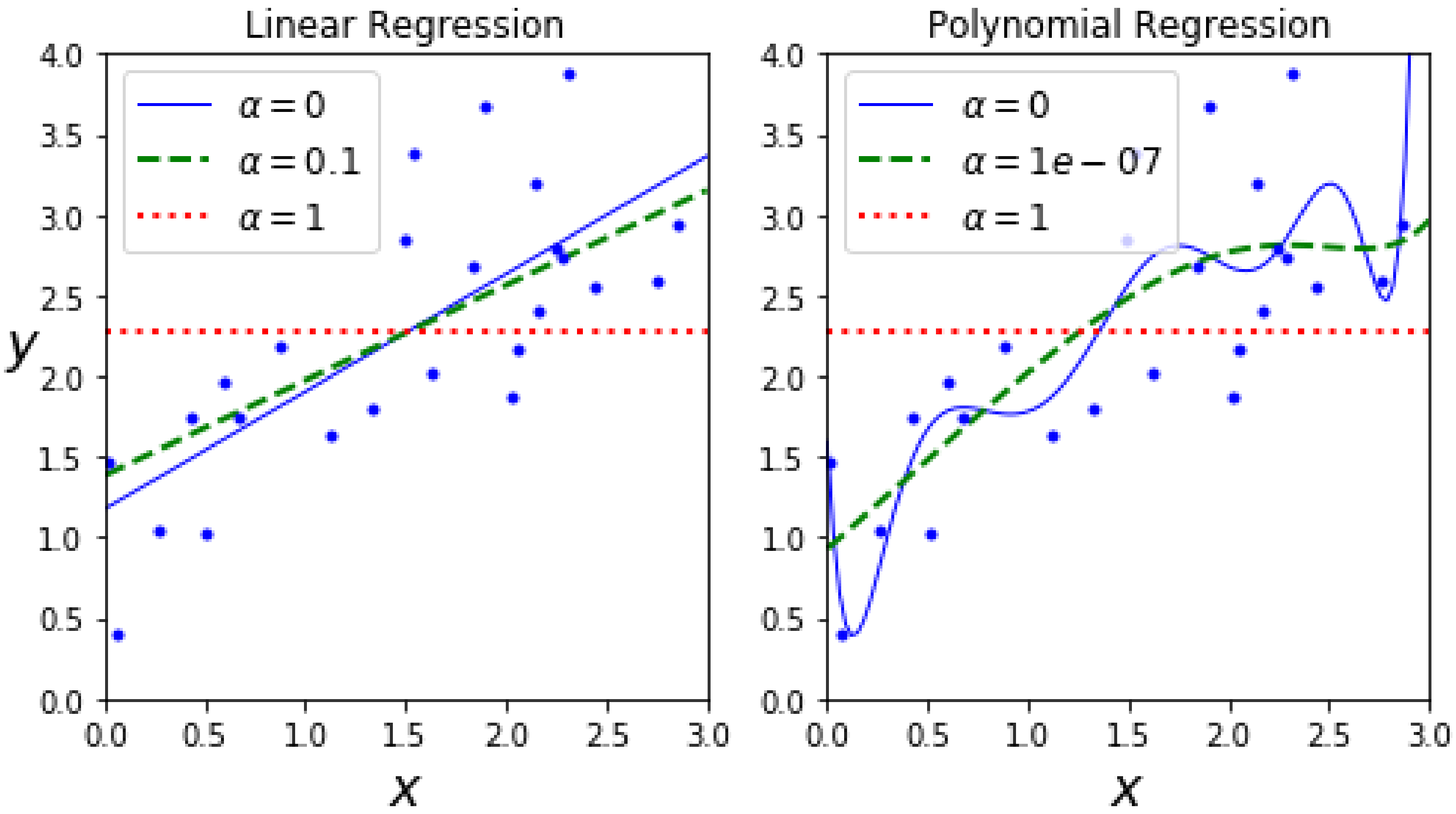
3) Constraint for LASSO regression is diamond:

$$|\theta_1| + |\theta_2| \leq User\ Defined\ Threshold$$

4) In constraint optimization, solution would lie at the active constraints (on the boundary of the constraint surface).
5) Unlike the disk, the diamond has corners; if the solution occurs at a corner, then it will make one parameter θ$_j$ equal to zero.
6) Thus, shrinkage in Lasso is more than in Ridge.

Elastic Net combines LASSO and Ridge Regression

LASSO Regression



Increase in α values flattens the curves and increases the bias

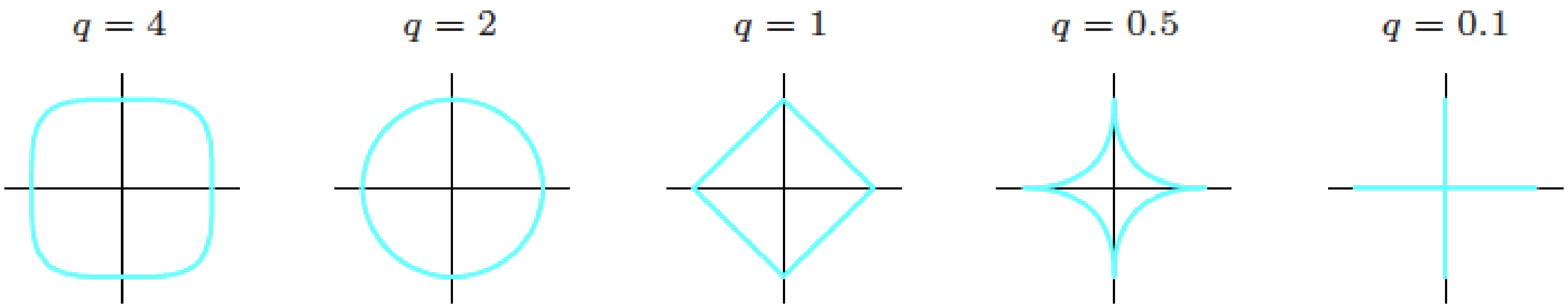
Generalization of LASSO

1) The objective function is modified in the generalized form of LASSO as follows:

$$\min J(\boldsymbol{\theta}) = MSE(\boldsymbol{\theta}) + \alpha \sum_{i=1}^{n} |\theta_i|^q$$

Regularization term

2) The value q = 0 corresponds to variable subset selection, as the penalty simply counts the number of nonzero parameters.
3) q = 1 corresponds to the Lasso.
4) q = 2 correspond to ridge regression.



$q = 4$ $q = 2$ $q = 1$ $q = 0.5$ $q = 0.1$

Elastic Net Regression

1) The elastic net is a regularized regression method that linearly combines the L_1 and L_2 penalties of the Lasso and Ridge methods.
2) The objective function is modified as follows:

$$\min J(\boldsymbol{\theta}) = MSE(\boldsymbol{\theta}) + r\alpha \sum_{i=1}^{n} |\theta_i| + \frac{1-r}{2} \alpha \sum_{i=1}^{n} |\theta_i|^2$$

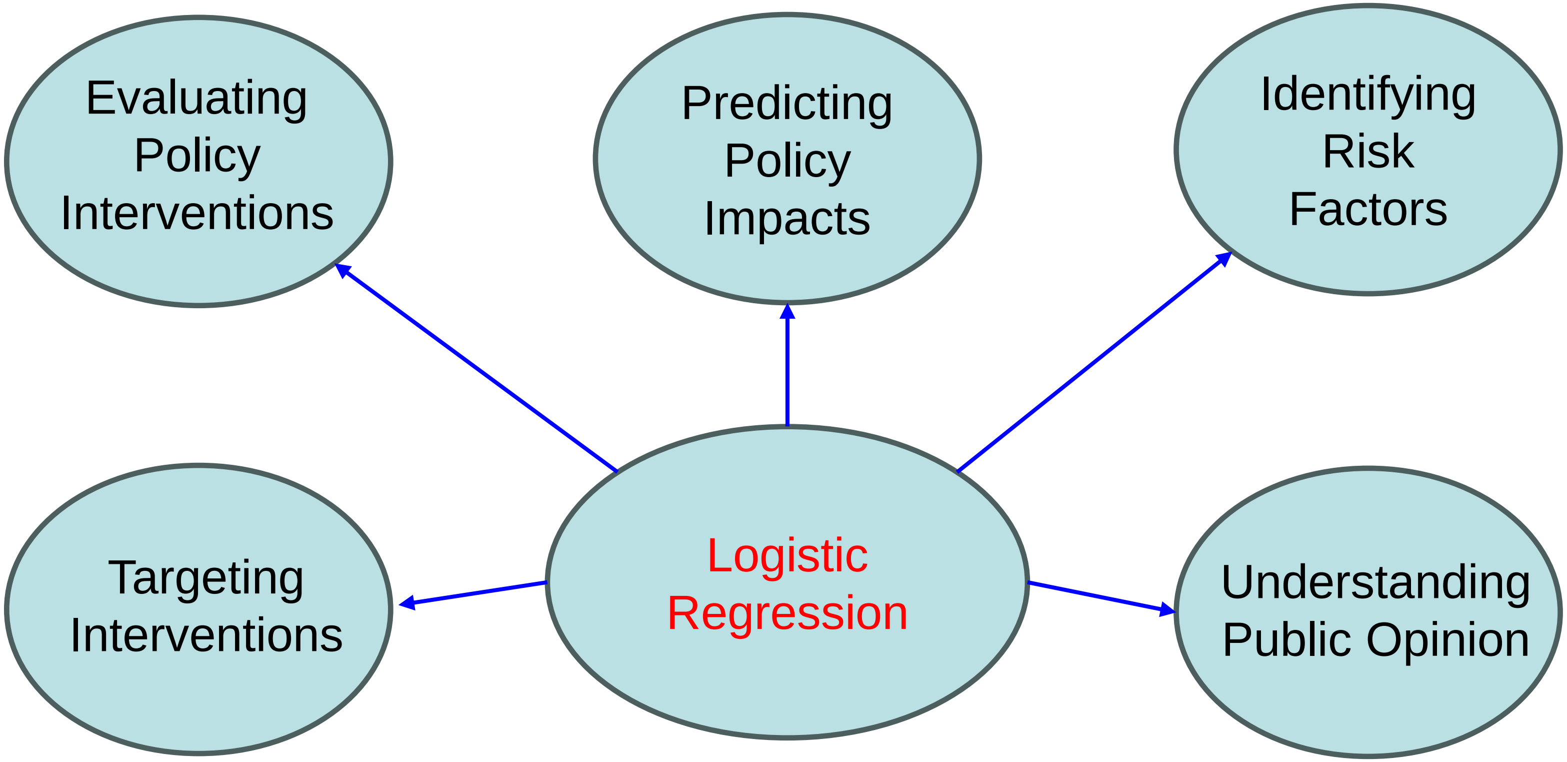
2) r = 1 corresponds to the Lasso.
3) r = 0 correspond to ridge regression.

Application of Logistic Regression

Impact of a new education program on the likelihood of high school graduation

Predict the probability of a student dropping out of school based on various demographic and socio-economic factors

Impact of substance abuse on probability of disease and insurance



Evaluating Policy Interventions

Predicting Policy Impacts

Identifying Risk Factors

Targeting Interventions

Logistic Regression

Understanding Public Opinion

Identify demographic groups that are at the highest risk of unemployment

Predict the likelihood of individuals supporting or opposing a particular policy based on their demographic characteristics, attitudes, and beliefs.

Case Study

Background: Diabetes is a significant health problem in India, posing challenges at both individual and societal levels. India's healthcare infrastructure faces challenges in effectively managing and preventing diabetes. Limited access to healthcare facilities, particularly in rural areas, poses barriers to timely diagnosis and treatment. Additionally, there's a shortage of healthcare professionals trained in diabetes management.

In this case study, a dataset involving 530 data points with the input features (Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, diabetes pedigree function, Age) and output feature [0 (no diabetes), 1(with diabetes)] is given. Complete data will be given in a CSV file.

Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age	Outcome
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
1	89	66	23	94	28.1	0.167	21	0
0	137	40	35	168	43.1	2.288	33	1
5	116	74	0	0	25.6	0.201	30	0

Case Study

Problem:

In this scenario, you're presented with a dataset containing crucial information on individuals' health parameters and whether the person has diabetes or not.

1) Plot mean values of features with and without diabetes.
2) Split the data into training and testing sets and perform the training using the maximization of the log-likelihood function.
3) Plot the confusion matrix for test results and evaluate the outcome.
4) Plot the decision boundary with respect two most important features.
5) Plot the relative importance of different features in predicting the output.

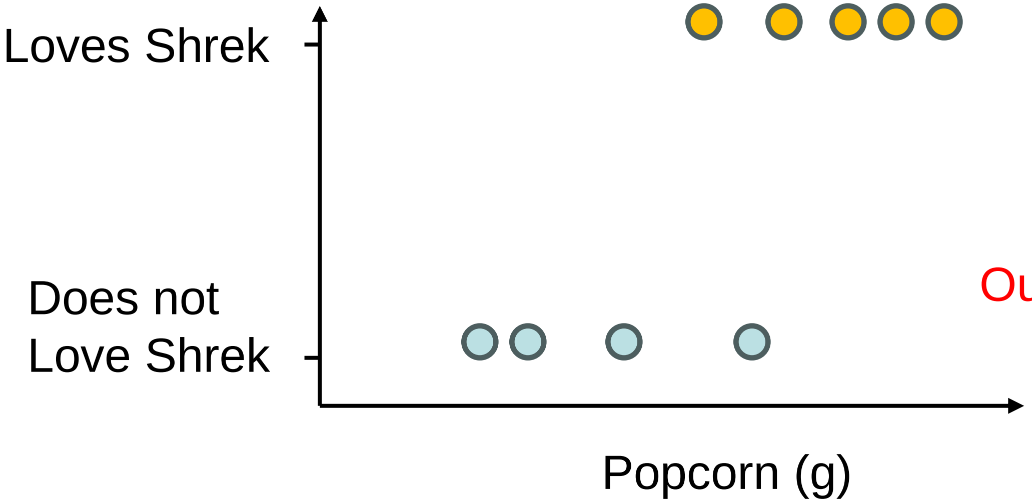
AIM: To understand the concept of logistic regression for binary classification problems.

Applications:

Developing the logistic regression model for predicting the output has potential application in evaluating the impact of different policies.

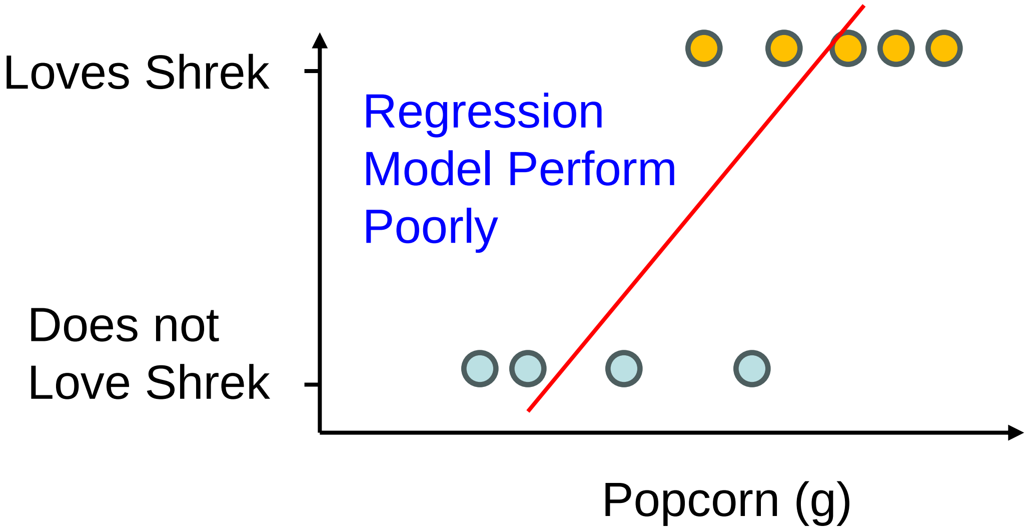
Logistic Regression

Logistic regression aims at solving binary classification problem in which output y can take on only two values, 0 and 1. Usual regression performs poorly for such cases.



Can we predict whether a person loved a movie Shrek based on how much Popcorn he/she eat?

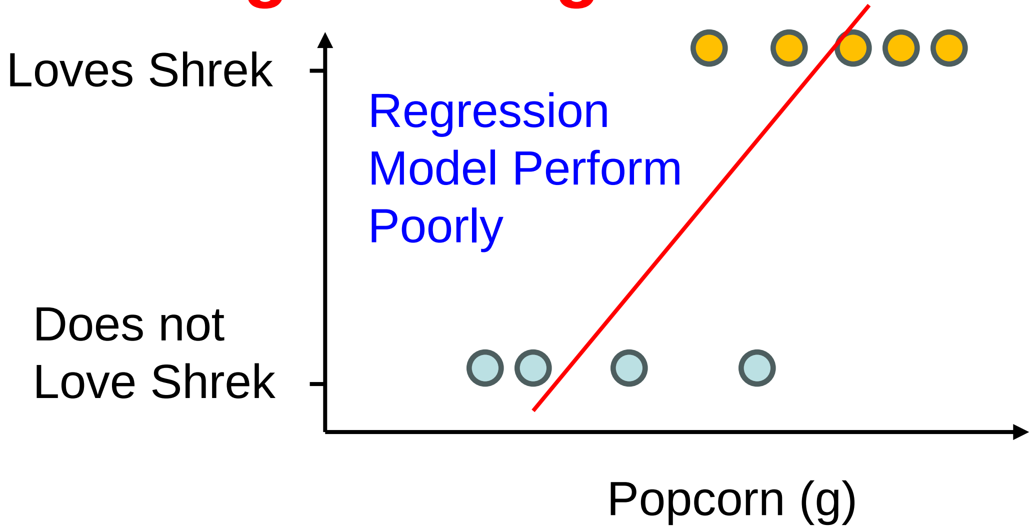
Output is categorical



Regression Model Perform Poorly

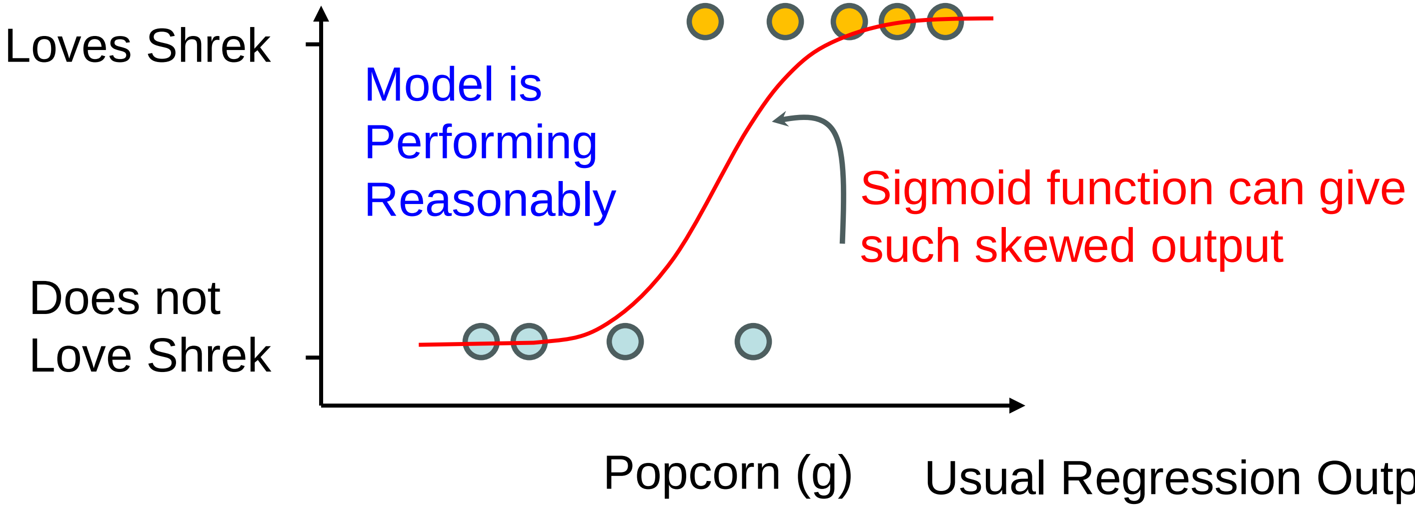
Regression is for continuous output and will perform poorly for discrete outputs

Logistic Regression



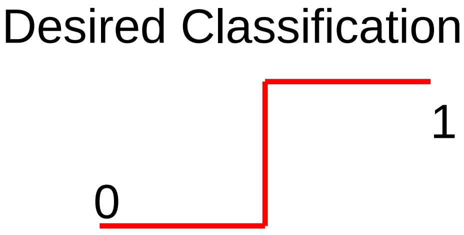
Regression Model Perform Poorly

Can we make Regression output skewed towards discrete values?



Model is Performing Reasonably

Sigmoid function can give such skewed output

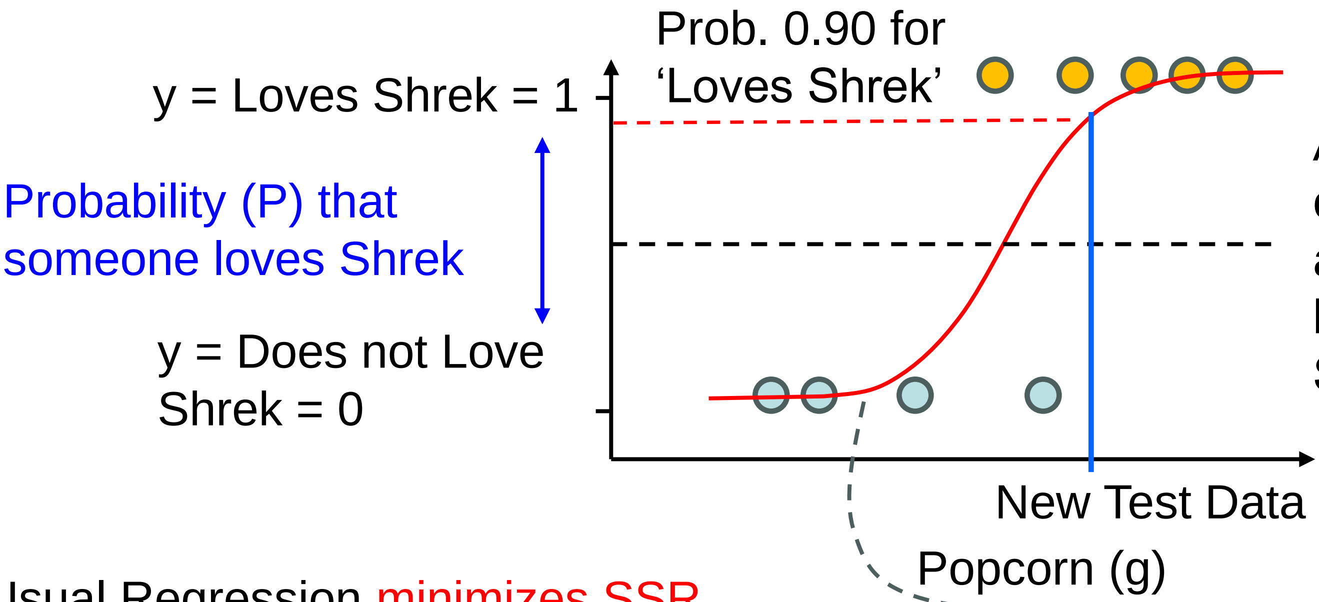


Usual Regression Output = $\theta_0 + \theta_1 \times \text{Feature}$

$$\text{Logistic Regression Output} = \frac{1}{1 + e^{-(Regression\ Output)}}$$

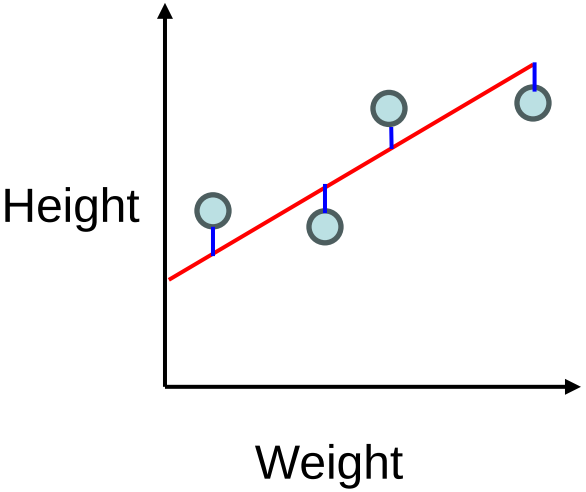
Sigmoid Function

Logistic Regression



Above this 0.5 threshold output will be classified as 'loves Shrek' and below 'does not love Shrek'.

Usual Regression minimizes SSR



Logistic Regression fits the sigmoid curve that maximizes the likelihood of correct prediction at all data points.

Likelihood of one data point for given x and θ = $P^{y}(1-P)^{(1-y)}$

Combined Likelihood for all data point with common x and θ = Likelihood1 × Likelihood1 ×…

Typically Log of Likelihood is taken as usual likelihood becomes smaller with each multiplication (underflow).

Logistic Regression

1) It is an interesting extension of regression concept which is primarily used for a prediction problem to a classification problem.
2) Closed form solution is not possible for logistic regression and numerical optimization methods are used for training the model.
3) The objective function used is maximization of log likelihood function.
4) Typically, the optimization method used is stochastic gradient ascent.
5) The concept of Bernoulli distribution is used for representing the likelihood function.

Logistic Regression

1) g(z) has an interesting property:

$$g'(z) = \frac{d}{dz}\frac{1}{1+e^{-z}} = -\frac{1}{\left(1+e^{-z}\right)^2}\left(-e^{-z}\right) = \frac{1}{\left(1+e^{-z}\right)}\frac{e^{-z}}{\left(1+e^{-z}\right)}$$

$$= \frac{1}{\left(1+e^{-z}\right)}\left(1-\frac{1}{\left(1+e^{-z}\right)}\right) = g(z)\left(1-g(z)\right)$$

2) Let us assume that:

$$p\left(y=1\middle|\mathbf{x};\boldsymbol{\theta}\right) = h_\theta(x)$$

$$p\left(y=0\middle|\mathbf{x};\boldsymbol{\theta}\right) = 1-h_\theta(x)$$

3) Combining these: $p\left(y\middle|\mathbf{x};\boldsymbol{\theta}\right) = \left(h_\theta(x)\right)^y\left(1-h_\theta(x)\right)^{(1-y)}$

4) MLE: If there are m independent examples:

$$L(\boldsymbol{\theta}) = p\left(Y\middle|X;\boldsymbol{\theta}\right) = \prod_{i=1}^{m} p\left(y^{(i)}\middle|\mathbf{x}^{(i)};\boldsymbol{\theta}\right) = \prod_{i=1}^{m}\left(h_\theta\left(\mathbf{x}^{(i)}\right)\right)^{y^{(i)}}\left(1-h_\theta\left(\mathbf{x}^{(i)}\right)\right)^{\left(1-y^{(i)}\right)}$$

Logistic Regression

1) Loglikelihood:

$$I\left(\boldsymbol{\theta}\right) = \ln L\left(\boldsymbol{\theta}\right) = \sum_{i=1}^{m}\left[y^{(i)} \ln h_\theta\left(\boldsymbol{x}^{(i)}\right) + \left(1-y^{(i)}\right)\ln\left(1-h_\theta\left(\boldsymbol{x}^{(i)}\right)\right)\right]$$

2) Gradient Ascent (as we are maximizing the Loglikelihood):

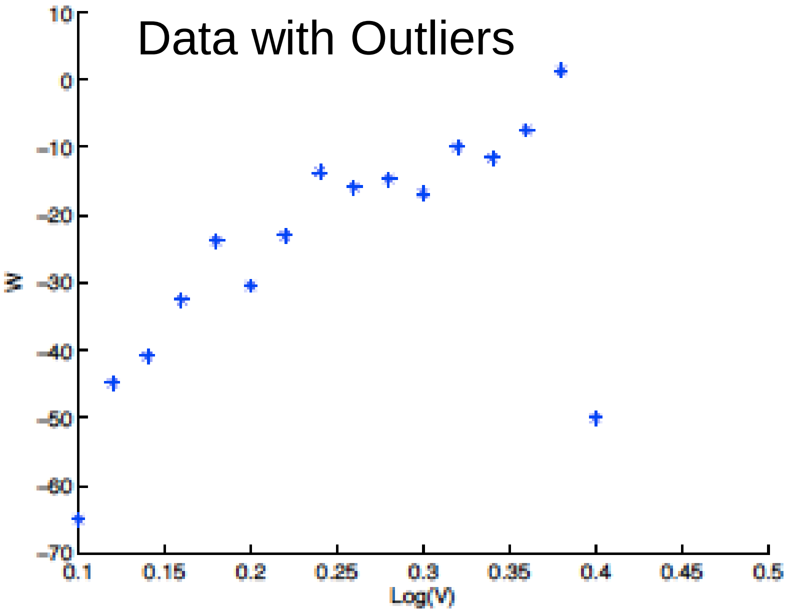
$$\frac{\partial}{\partial\theta_j} I\left(\boldsymbol{\theta}\right) = \left(y\frac{1}{g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)} - (1-y)\frac{1}{\left(1-g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)\right)} \right)\frac{\partial}{\partial\theta_j} g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)$$

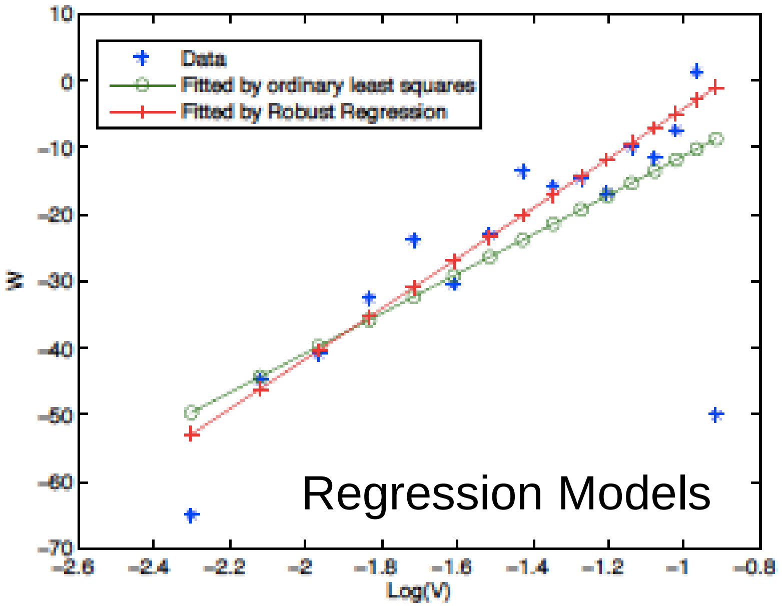
$$= \left(y\frac{1}{g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)} - (1-y)\frac{1}{\left(1-g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)\right)} \right) g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)\left(1-g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)\right)\frac{\partial}{\partial\theta_j}\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)$$

$$= \left(y\left(1-g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)\right) - (1-y)\,g\left(\boldsymbol{\theta}^T\boldsymbol{x}\right)\right)x_j = \left(y - h_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right)\right)x_j$$

$$\theta_j\Big|_{k+1} = \theta_j\Big|_k + \alpha\left(y - h_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right)\right)x_j\Big|_k$$

Robust Regression


Data with Outliers


Regression Models

1) Often the data has outliers. In such cases, the usual least square regression is dominated by these outliers which results in inaccurate model prediction.
2) To regulate the effect of such outlier points, the concept of robust regression (Huber's Regression) is used.

$$J(\theta) = \sum_{i=1}^{m} g(\in)^{(i)}; g(\in) = \begin{cases} \frac{1}{2}\in^2 & if\ |\in| \le M \\ M\left(|\in| - \frac{1}{2}M\right) & if\ |\in| > M \end{cases}$$

where, ∈ is the error at each data point, and M is the threshold tolerance for outliers.

Case Study

Background: The government of India has implemented various schemes aimed at providing financial assistance and credit facilities to citizens, particularly those in the micro, small, and medium enterprises (MSMEs) sector, as well as to individuals in need of funding for entrepreneurial ventures or income-generating activities. One prominent scheme in this regard is the Pradhan Mantri Mudra Yojana (PMMY).

In this case study, a dataset involving 615 data points with the input features (Gender, Married, Dependents, Education, Self-Employment, Applicant Income, Co-applicant Income, Loan Amount, Loan amount term, Credit History, Property) and output feature [Loan Status: Y (granted), N (not granted)] is given. Complete data will be given in a CSV file.

Gender	Married	Dep.	Edu.	Self-Emp	App. Inc.	Co-app. Inc.	Loan amount	Term	Cred. Hist	Property	Loan Status
Male	Yes	1	Graduate	No	4583	1508	128	360	1	Rural	N
Male	Yes	0	Graduate	Yes	3000	0	66	360	1	Urban	Y
Female	No	0	Graduate	No	3510	0	76	360	0	Urban	N
Male	No	0	Graduate	No	6000	0	141	360	1	Urban	Y

Case Study

Problem:

In this scenario, you're presented with a dataset containing crucial information on individuals.

1) Split the data into training and testing sets and perform the training using the maximization of the log-likelihood function.
2) Plot the confusion matrix for test results and evaluate the outcome.
3) Plot the decision boundary with respect two most important features.
4) Plot the relative importance of different features in predicting the output.

AIM: Such a predictive model can be used for rolling the targeted schemes.

Thank You

Linear Regression

2) Typically, m data points are present (each data point is a row with y, x_1, x_2, …, x_n in a Table) such that m > n.

Sample No.	**Actual output,** y	**First Feature,** x_1	**Second Feature,** x_2	**Third Feature,** x_3	‥	**n^{th} Feature,** x_n
1	$y^{(1)}$	$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	‥	$x_n^{(1)}$
2	$y^{(2)}$	$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	‥	$x_n^{(2)}$
:	:	:	:	:	:	:
m	$y^{(m)}$	$x_1^{(m)}$	$x_2^{(m)}$	$x_3^{(m)}$	‥	$x_n^{(m)}$

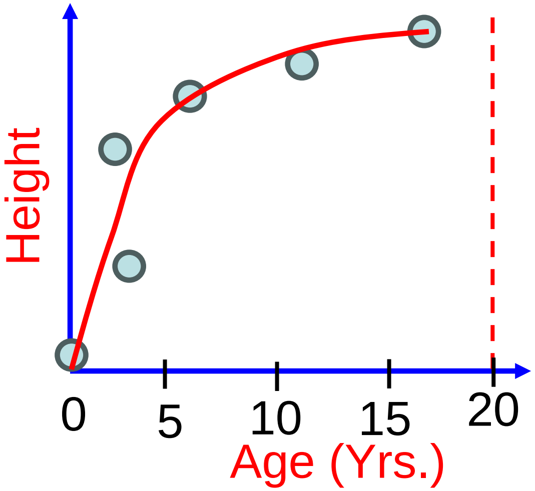
y X

3) Objective is to obtain the optimal values of weights, **θ**, which gives the best fit of the model output y' with given output y. Clearly, the error (residual) between y and y' need to be minimized. Typically, the **sum of square residual (SSR)** is minimized and the optimum RMSE or MSE is reported.

$$\min MSE(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m}\left[\overset{\text{Actual y}}{y^{(i)}} - \overset{\text{Predicted y}}{\left(\theta_0 x_0^{(i)} + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} + \cdots + \ \theta_n x_n^{(i)}\right)}\right]^2$$

Polynomial Regression

1) Often, the data is complex than that represented by a straight line. Fitting such data using a polynomial is referred as Polynomial regression.
2) Interesting to note that the linear regression can be used for representing such complex data by representing the 'feature with powers' by new features and then taking a linear combinations of these.



$$y' = \theta_0 + \theta_1 z + \theta_2 z^2 + \cdots + \ \theta_n z^n$$

The predicted output is represented in terms of power of feature z i.e., height is represented as polynomial of Age

To convert the problem to linear regression: $x_1 = z, x_2 = z^2, x_3 = z^3, \cdots, x_n = z^n$

$$y' = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

3) A polynomial of any degree for given number of features can be given as input to fit the data. For instance, if degree = 3 and features = 2 (a, and b) then the output will be

$$y' = \theta_0 + \theta_1 a + \theta_2 b + \overset{\overset{\theta_3 x_3}{\uparrow}}{\theta_3 ab} + \overset{\overset{\theta_4 x_4}{\uparrow}}{\theta_4 a^2 b} + \theta_5 ab^2 + \theta_6 a^3 + \theta_7 b^3$$

Features are correlated

4) Challenge is to obtain correct degree of the polynomial which do not cause underfitting or overfitting.

Logistic Regression

1) Logistic regression aims at solving binary classification problem in which output y can take on only two values, 0 and 1.
2) Since linear regression gives continuous valued output, it performs poorly for such classification problem in which desired output is discrete.

Logistic Regression Model:

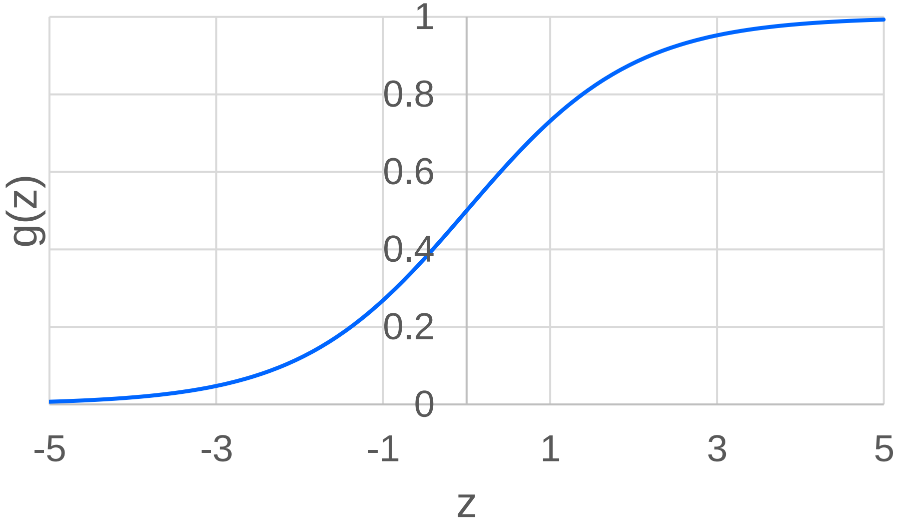
$$h_\theta(\boldsymbol{x}) = g\left(\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x}\right) = \frac{1}{1+e^{-\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x}}}$$

$$\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{x} = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Where the function $g(z)$ is given as:

$$g(z) = \frac{1}{1+e^{-z}}$$

is called logistic or sigmoid function



When z → ∞, g(z) → 1
When z → -∞, g(z) → 0

Sigmoidal function gives classification close to desired binary classification

Desired Classification: