Supervised Learning

California Science and Technology University

About Me

- Author and Technologist
- Worked for TI, Magma, Apache, Cadence, Paripath and now AITS.
- 20 years n EDA/CAD/ML industry

- www.linkedin.com/in/srohit0/
- medium.com/@srohit0
- qr.ae/TWGSt9
- github.com/srohit0
- twitter.com/srohit

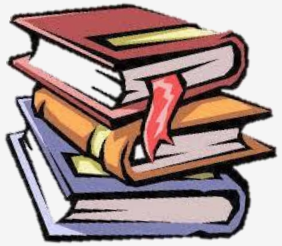
Course Overview

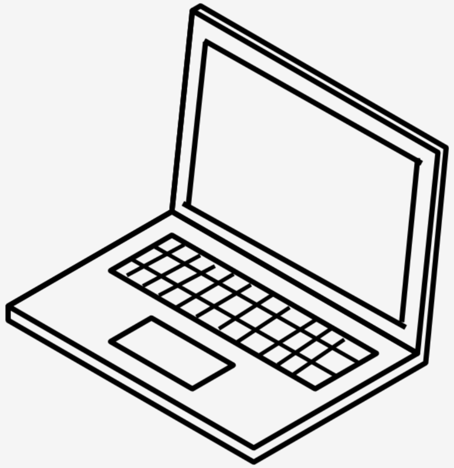
- Pre-requisites
  - basic computer science principles and skills
  - Probability theory
  - Multivariable calculus and Linear algebra
- Applied course with emphasis on real life projects
- Math and programming makes it fun and challenging
- Make friends for study groups for projects.
- Reference Book :
  - Machine Intelligence, Rohit Sharma, 2018.
  - srohit0.github.io/mida/

Homework	Quiz	Midterm Project	Final Project	Final Exam	Participation	Total
5%	15%	20%	25%	30%	5%	100%

Material

- Text Book:
  - Machine Intelligence by Rohit Sharma
- References Material:
  - Python Machine Learning, 2nd Edition, by Sebastian Raschka
  - Deep Learning, by Ian Goodfellow
- Software
  - Python
  - Google Colab or Jupyter notebook





Mid Term Projects ideas

- K Nearest Neighbor (dnnc)
- K means algorithm (dnnc)
- Decision Tree (dnnc)
- Random Forests (dnnc)
- Sales Forecasting using Walmart Dataset (sklearn, dnnc)
- MNIST training (sklearn, dnnc)
- MNIST inference (dnnc)
- MNIST model compilation (dnnc)
- Anomaly Detection (dnnc)
- Your suggestions.

KNN using DNNC

KNN Algorithm

Constructor:

1. load iris dataset
2. Initialize variables

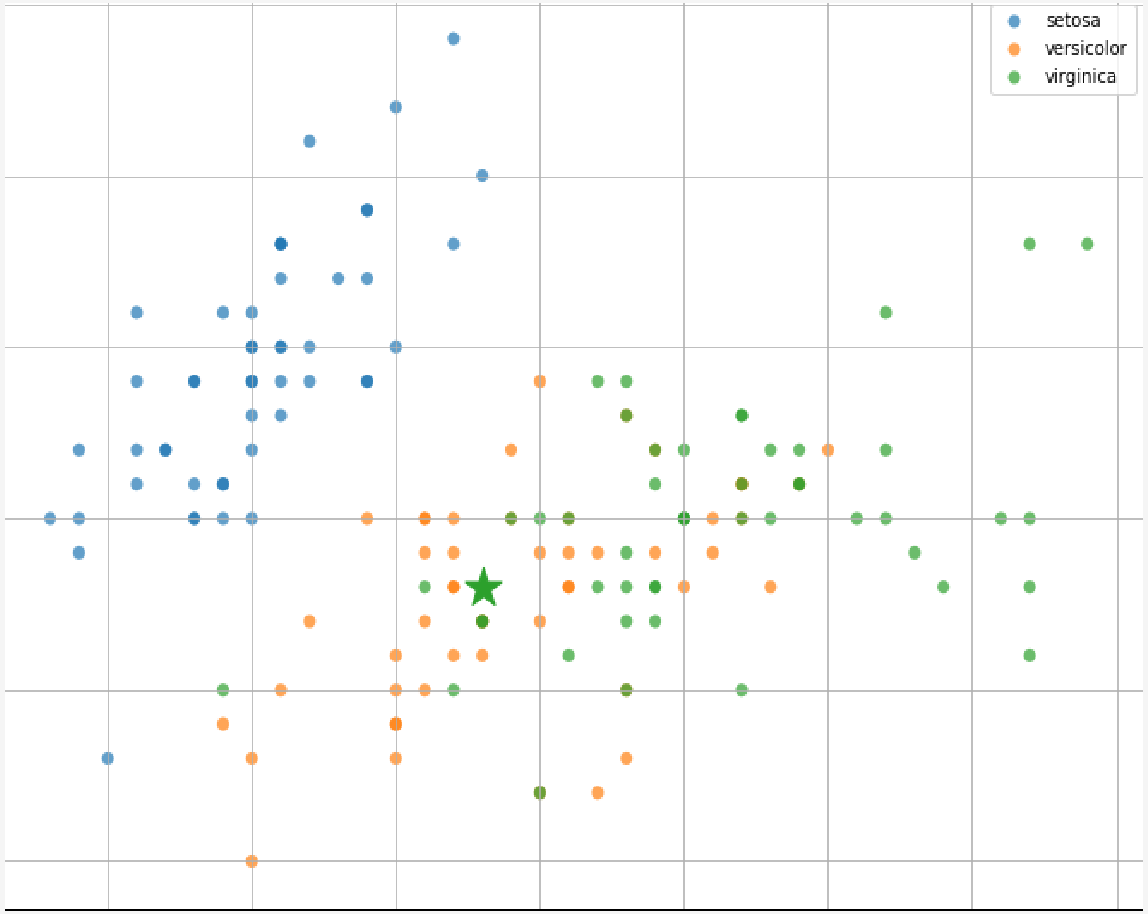
Fit

1. Survey opinions in the neighorhood
2. compute difference of opinions using euclidean distance, i.e.

   $\sum_i \sqrt{(feature_i - query)^2}$
3. sort server based on increasing order of euclidean distance.
4. count top-K votes.

Predict

1. return majority vote, i.e number of labels in top-k votes.



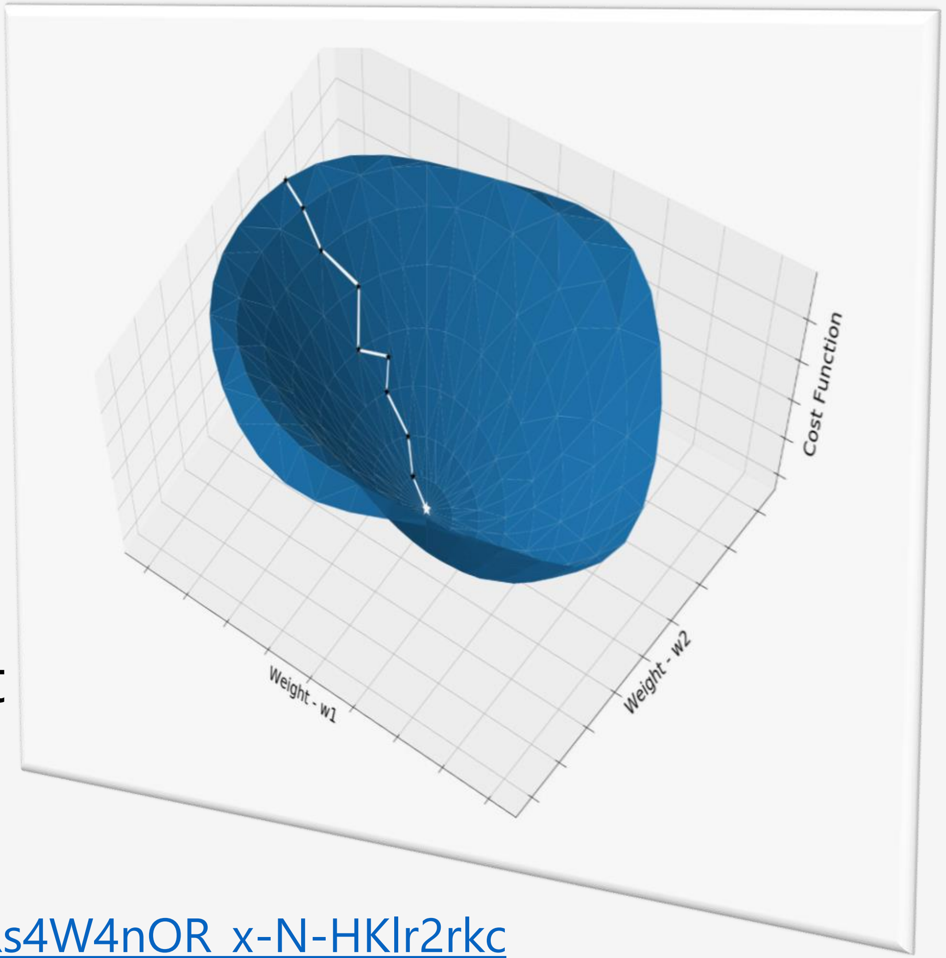
Review - Learning Algorithm

- Learning Algorithm minimizes loss function to find appropriate values for the model within constraints
- Steps of the algorithm are given below:

repeat until convergence:

$$w_{i+1} := w_i - \alpha \nabla J(w_i)$$

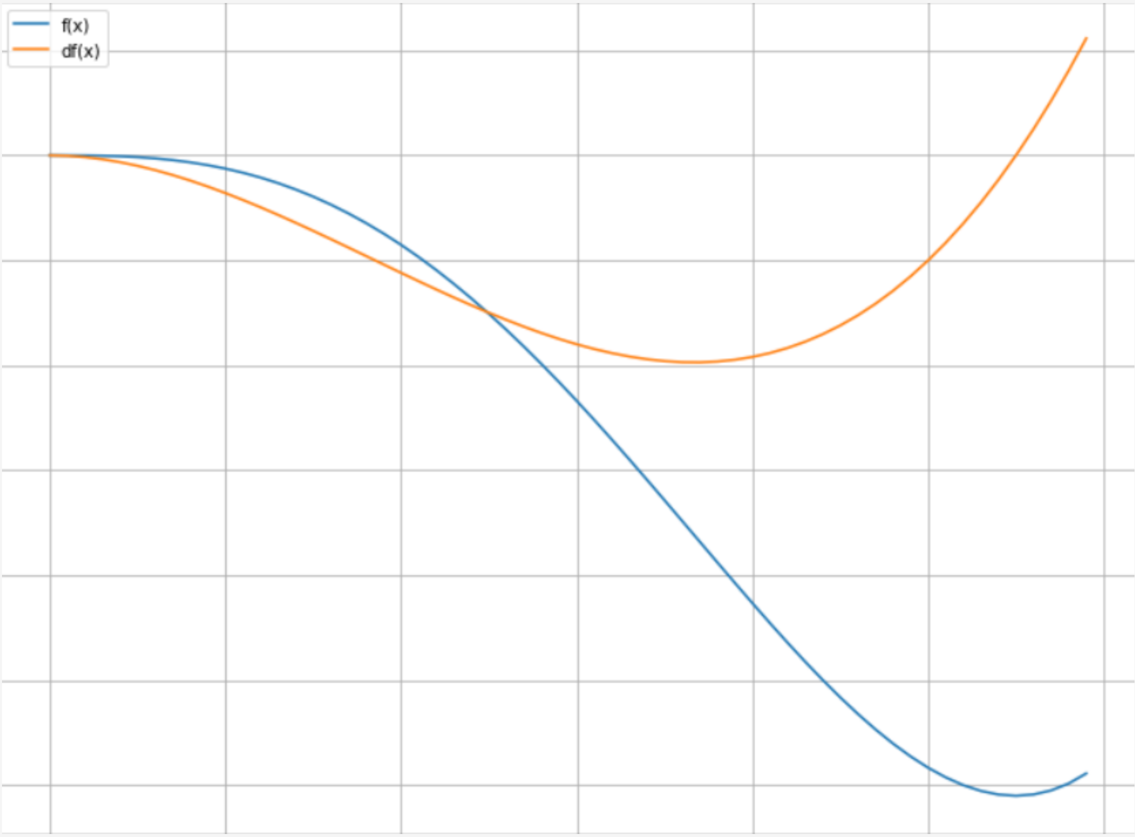
- Here ∇ is a vector of partial derivatives of cost function



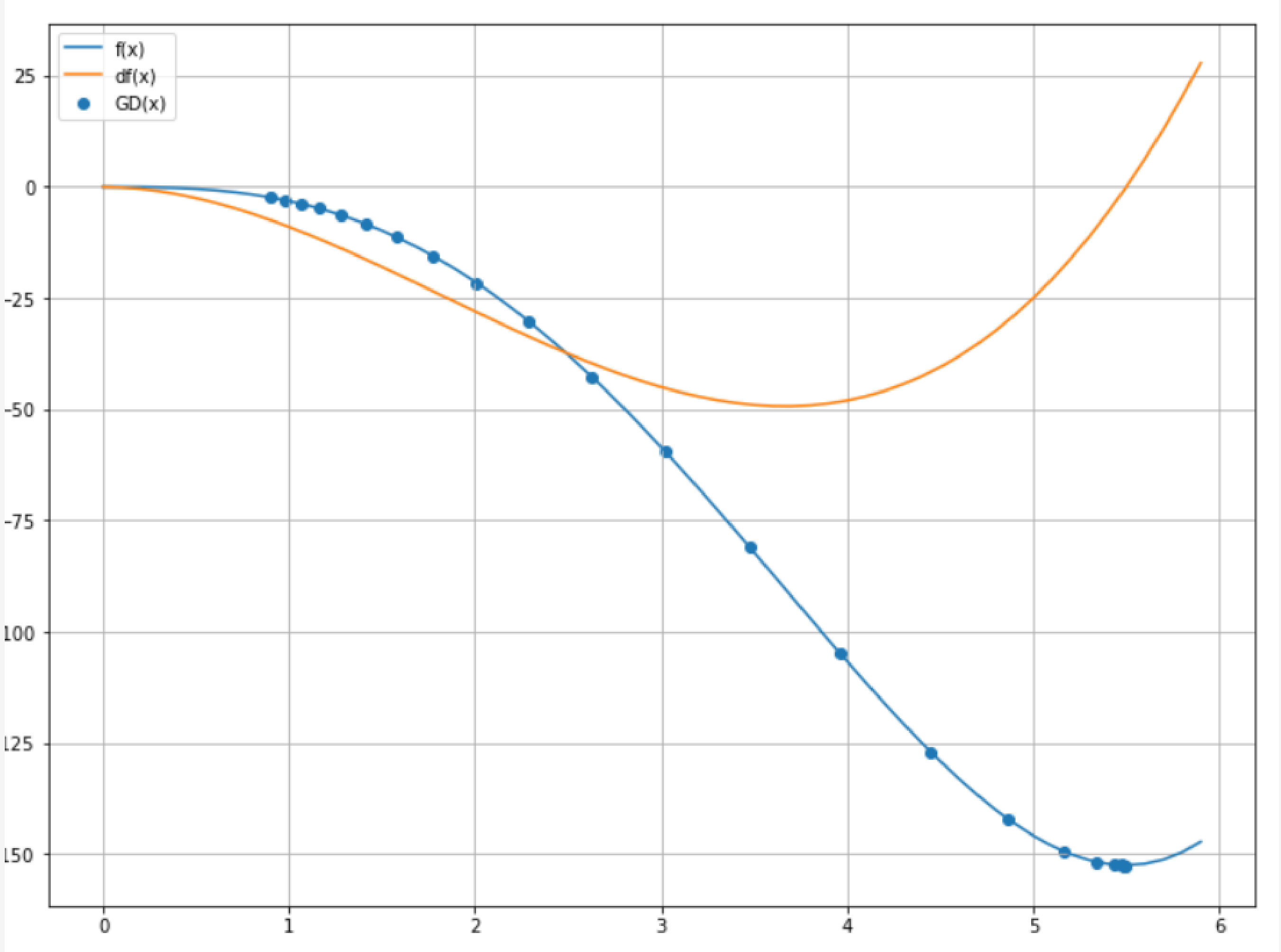
https://colab.research.google.com/drive/1a6kmnGs8McbRs4W4nOR_x-N-HKlr2rkc

Gradient Descent

Learning Rate α	Iterations	final value	accuracy
1e-1	infinite	NA	NA
8e-2	5	0.0483	fair
1e-2	22	5.499	excellent
1e-3	208	5.48	good
1e-4	1875	5.45	fair
5e-5	3633	5.42	fair
2e-5	8667	5.37	ok
1e-5	NAN	NA	NA

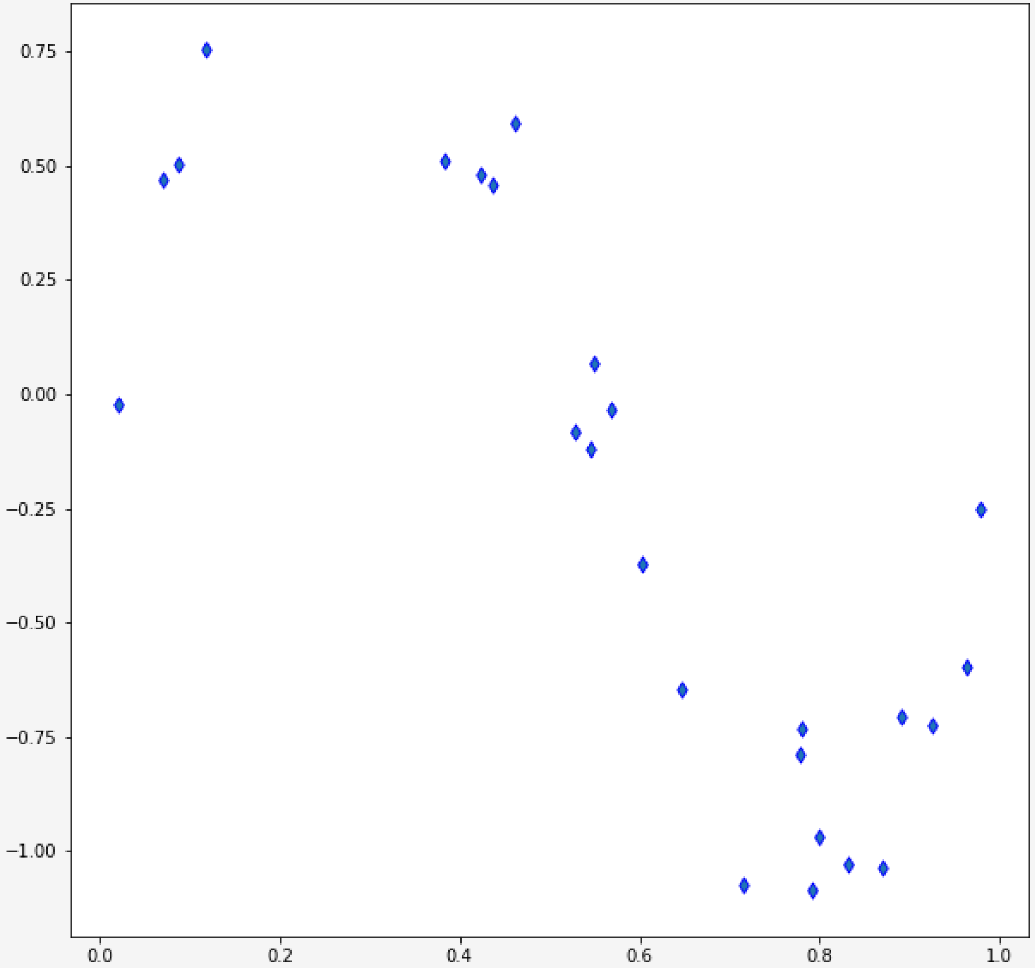


Gradient Descent Process



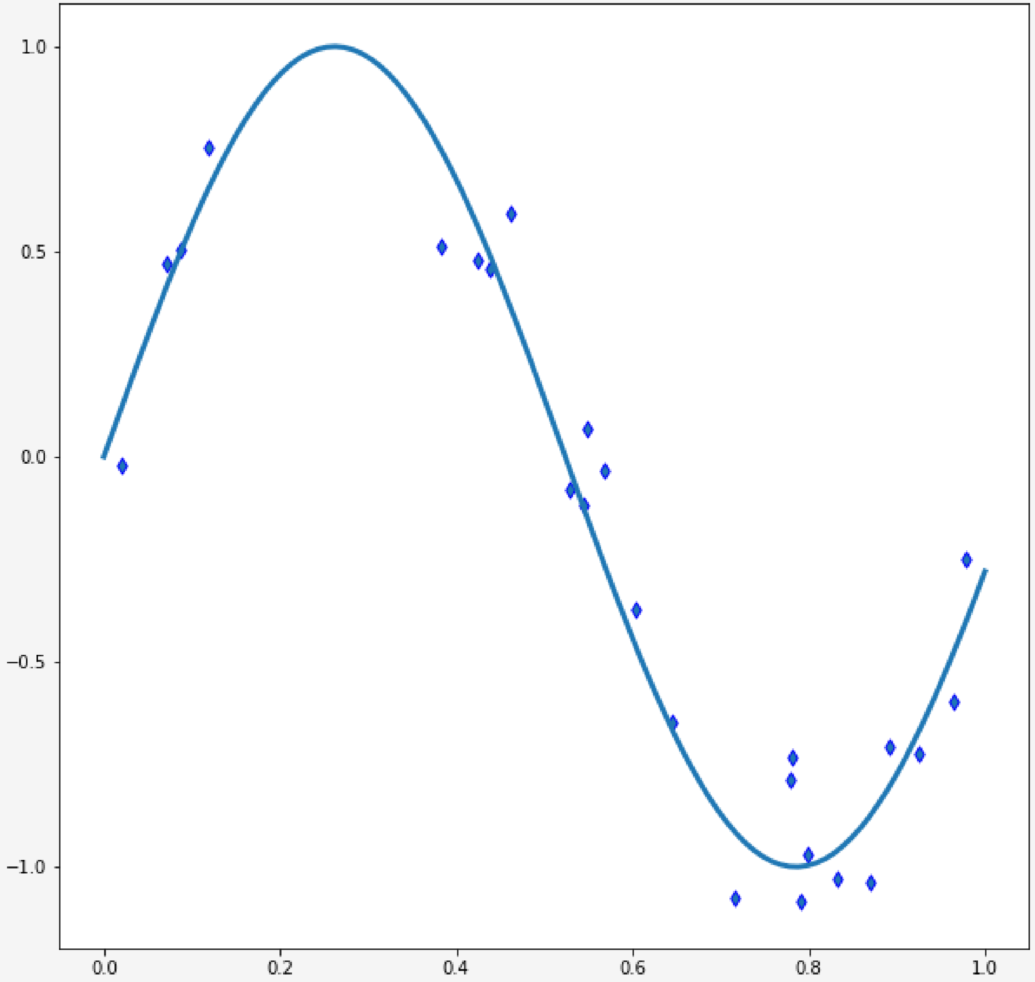
Synthetic Model

Guess the model?



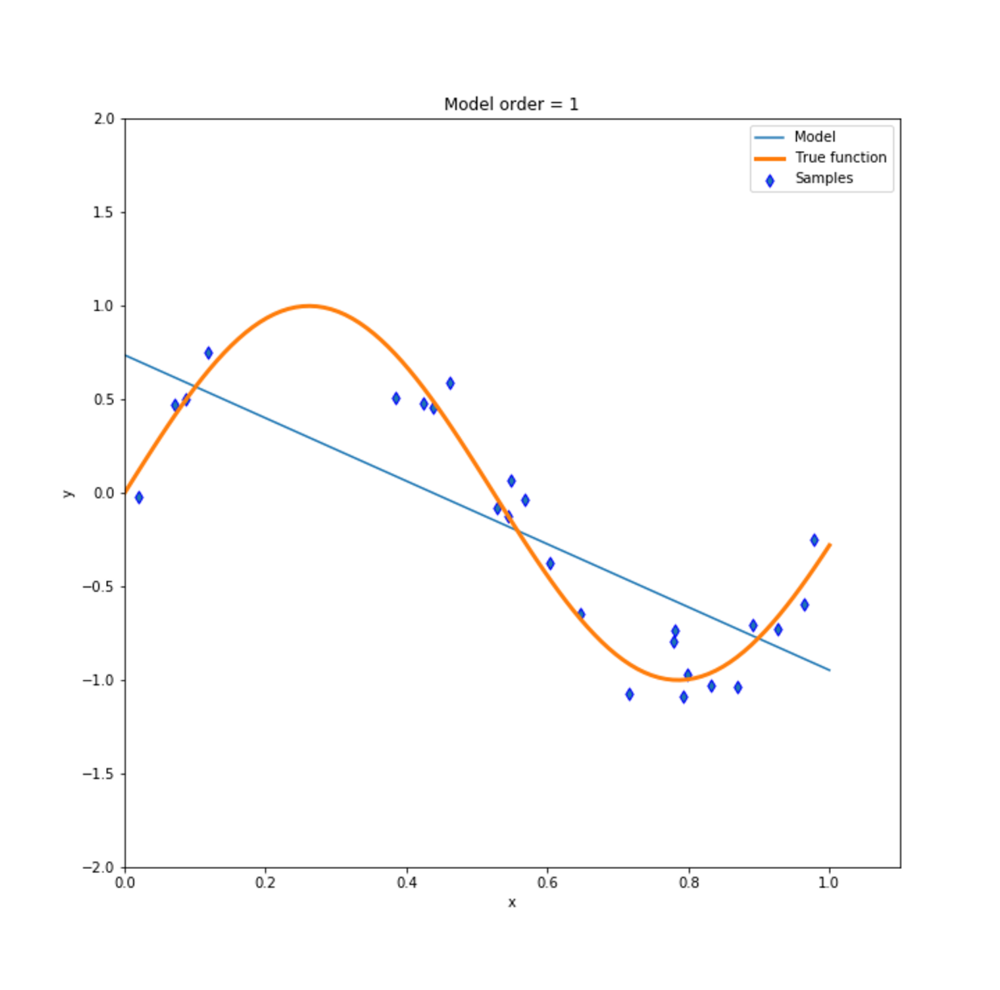
Synthetic Model

Y(x) = sin(6x)
With 15% random noise



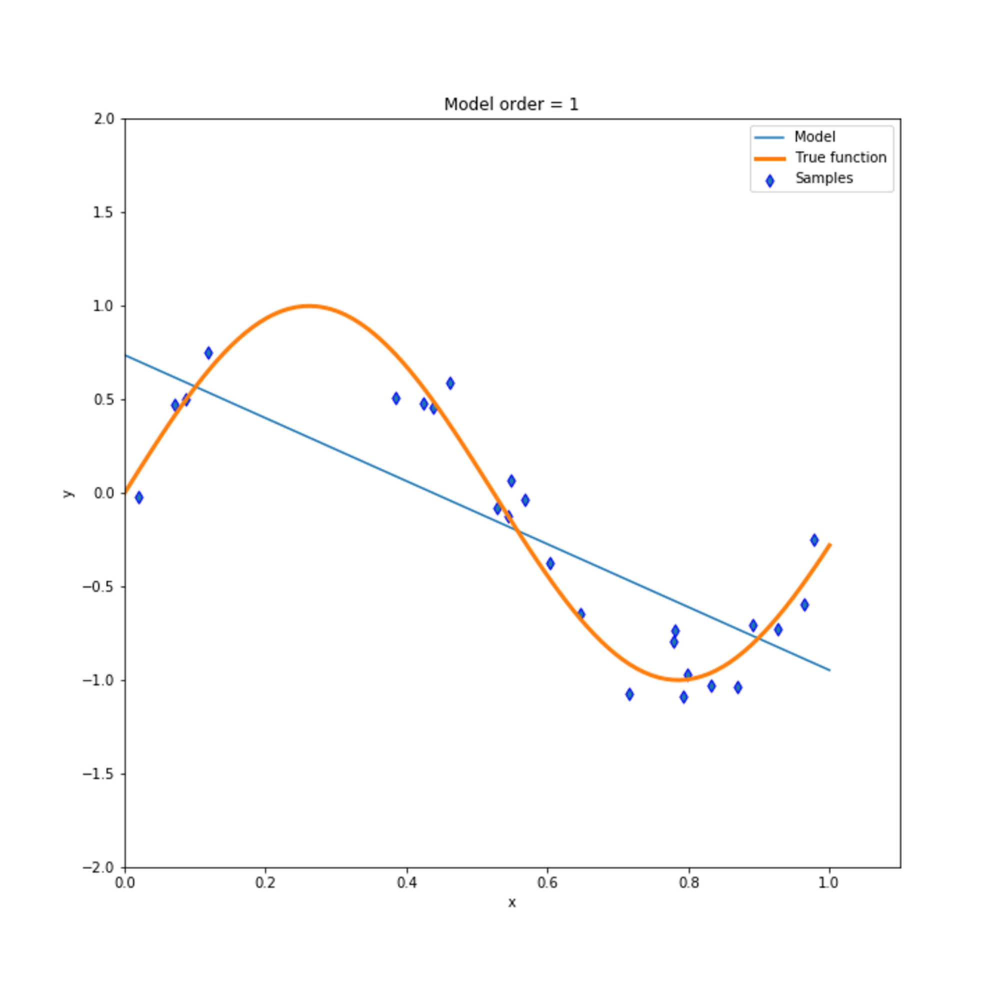
Underfitting w/ linear model

- Underfitting is said to occur when the model fails to learn dataset attributes and does not fit training dataset and validation dataset.
- Model evaluation returns high cost, poor accuracy with a high bias.



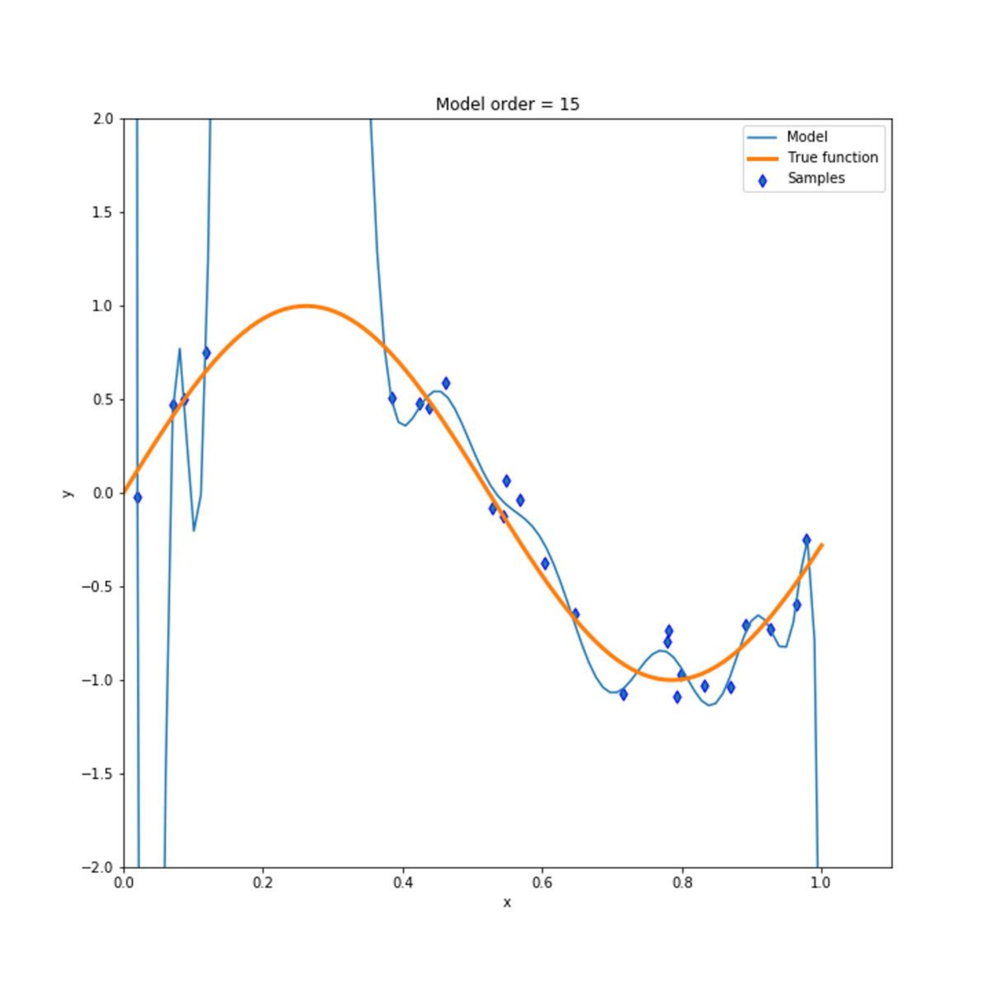
Addressing Underfitting

1. Train for a longer duration, if cost is consistently reducing.
2. Reduce learning rate, if training has wild fluctuation.
3. Change the model
    1. Add more parameters in the model
    2. Increase order of the model.
4. Try a new model (e.g., linear to polynomial or deep learning)



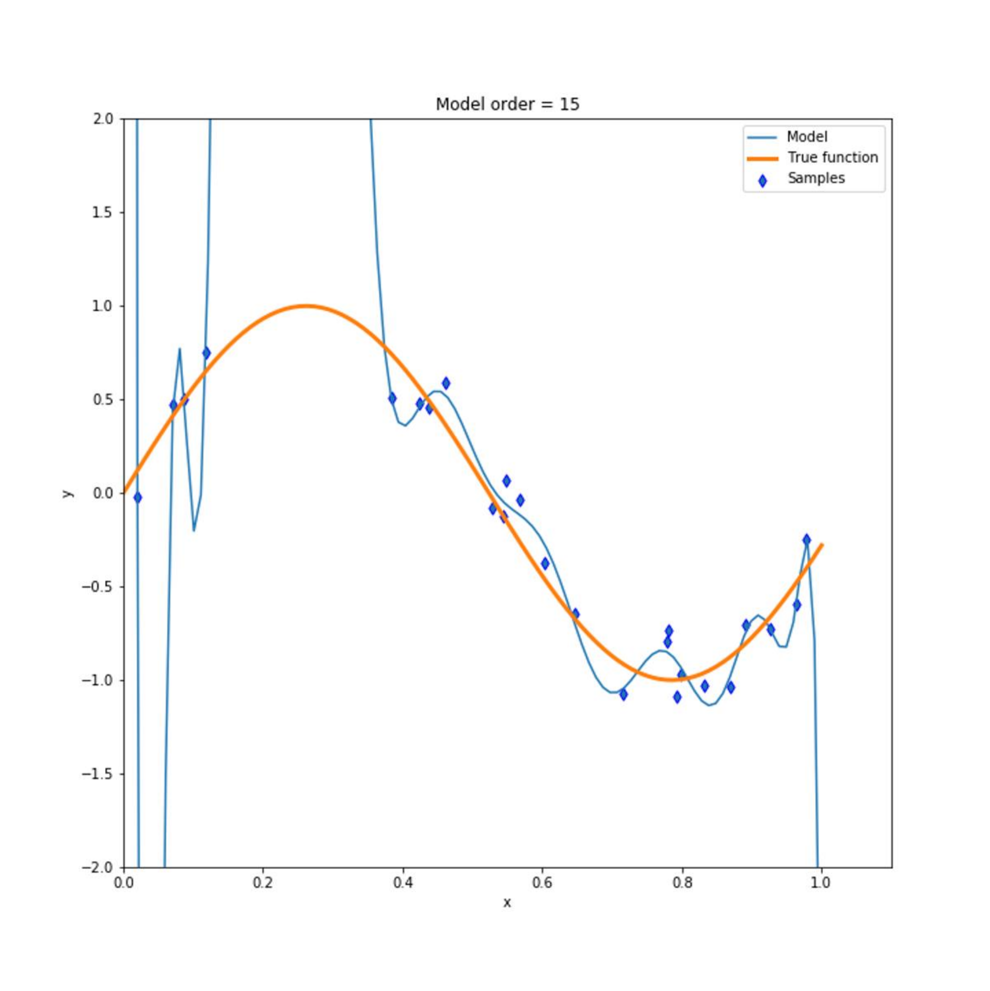
Overfitting w/ Order 15 model

- Overfitting results, when model learns patterns and attributes of dataset with high variance along with those of noise and outliers. It achieves good accuracy on the training set and performs poorly on validation set.

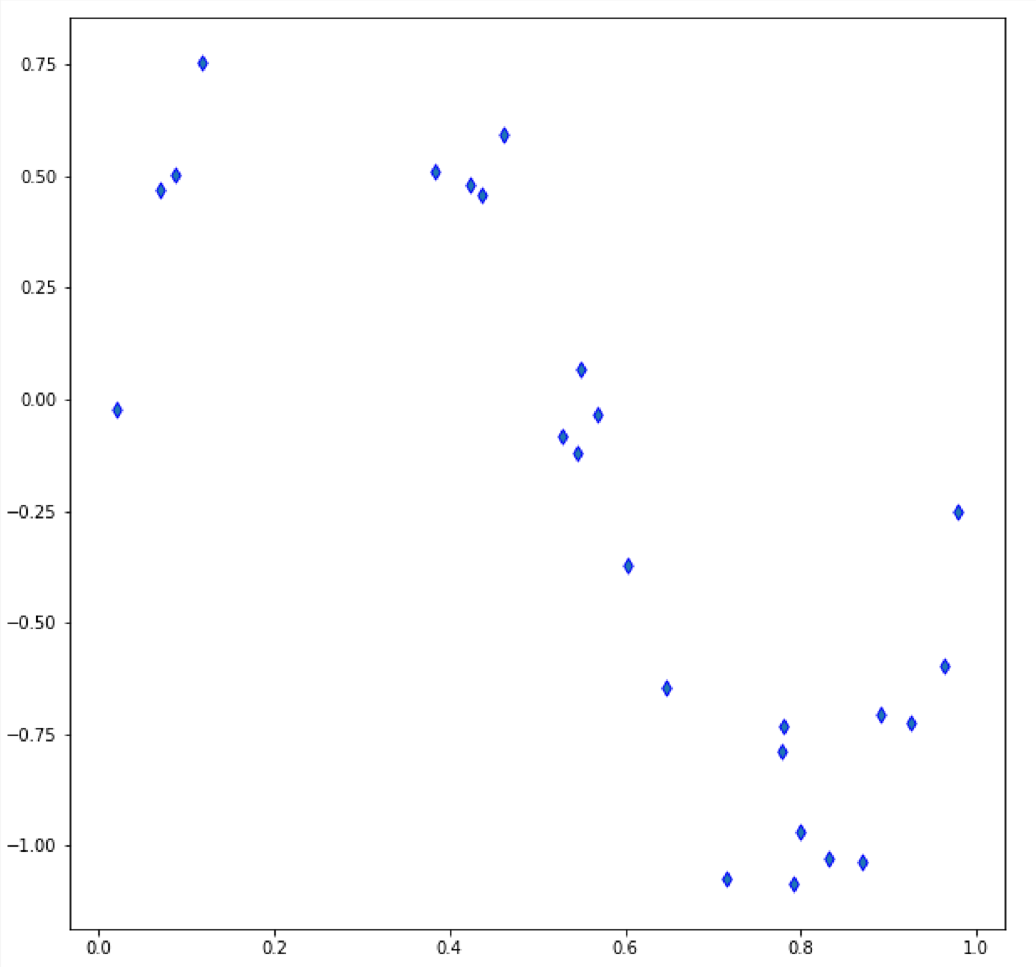


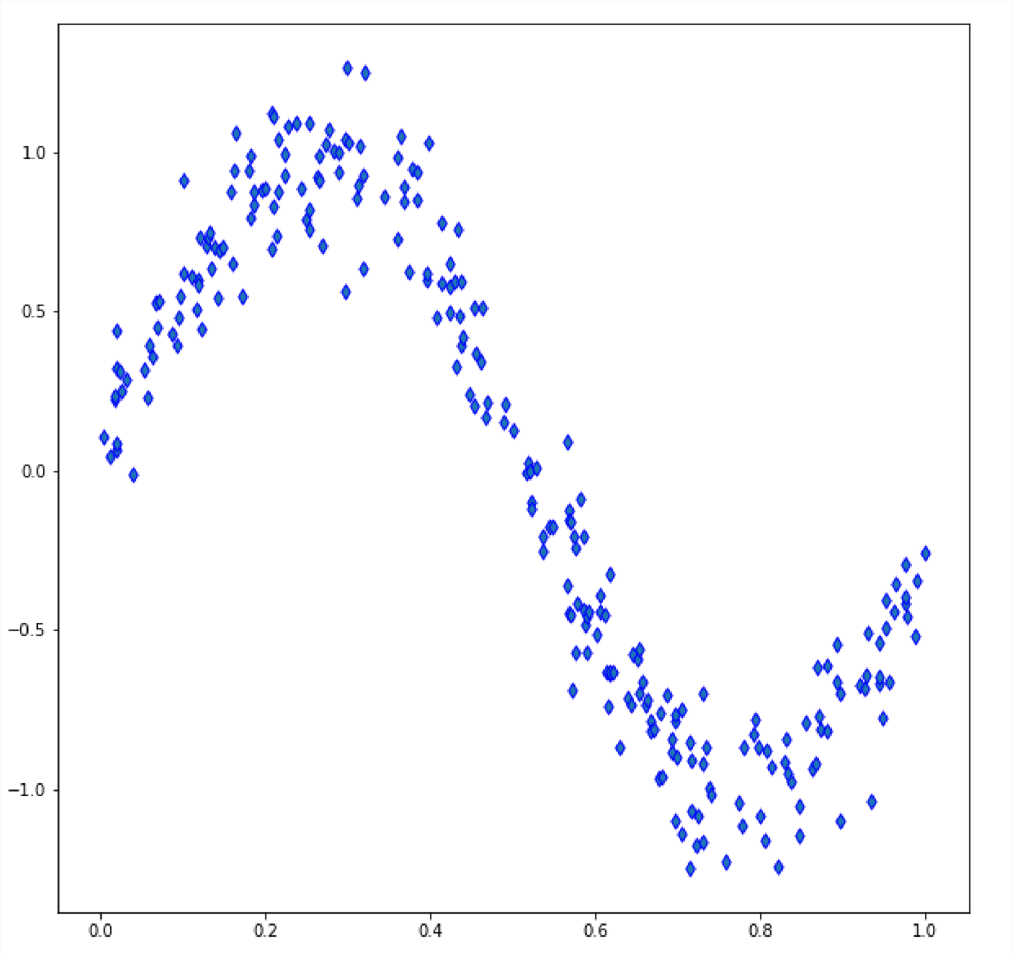
Overfitting w/ Order 15 model

- The model becomes too specific to training data and fails to generalize the learning to validation set. The problem originates from false variation resulting from dataset noise learnt as a concept in the model. Fixes include:
  - Simplify model
  - Collect more data
  - Use techniques like regularization and dropout

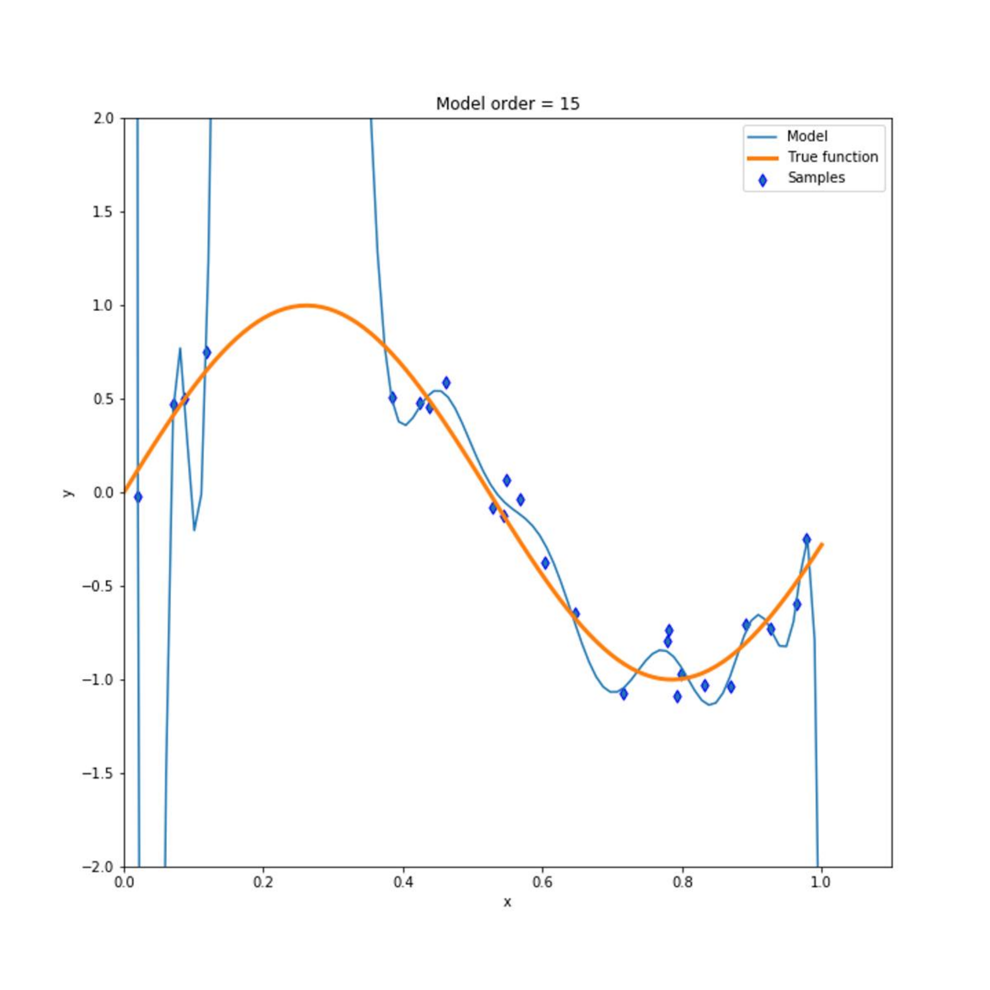


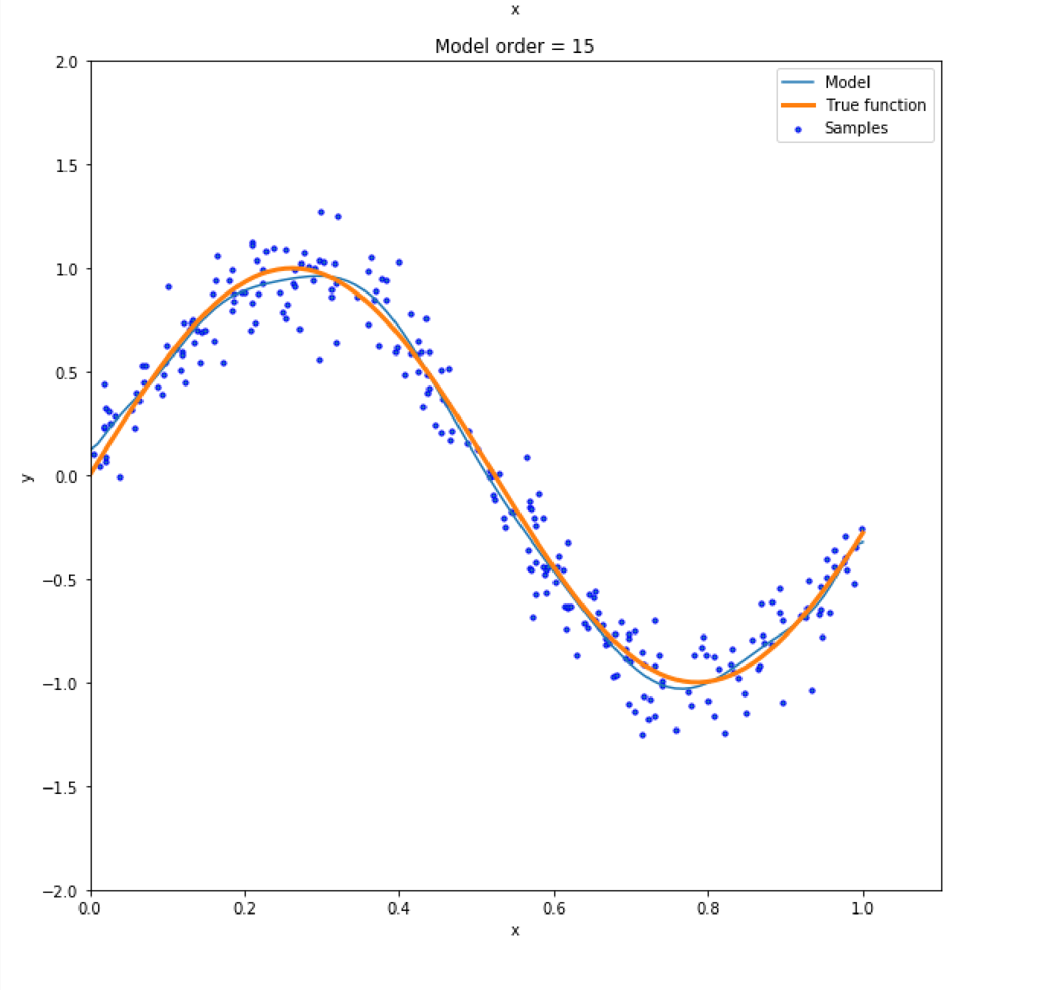
What if we increase dataset size



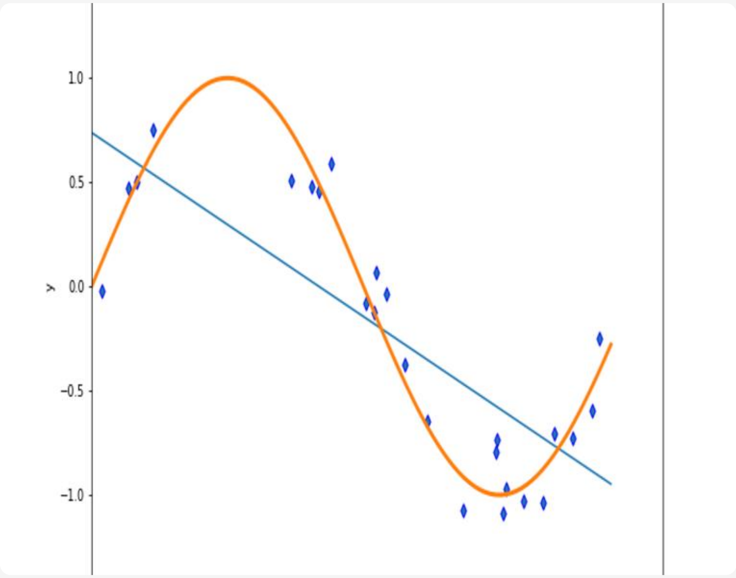


Effect of increasing dataset size

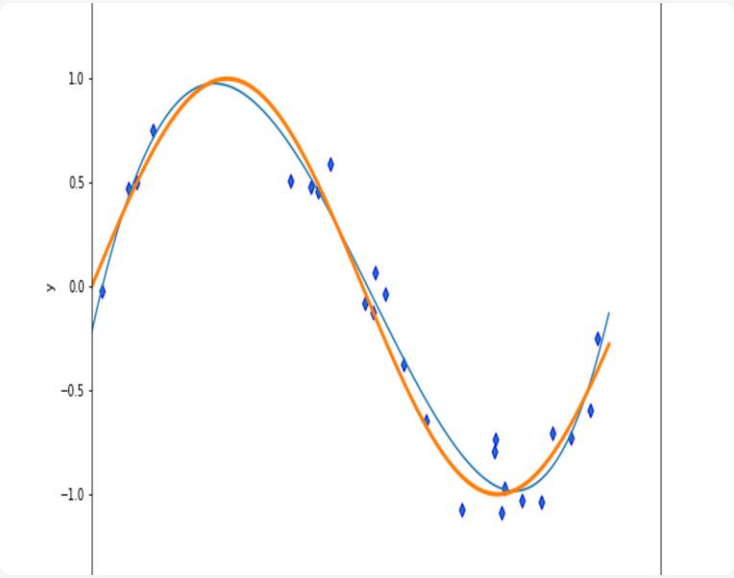




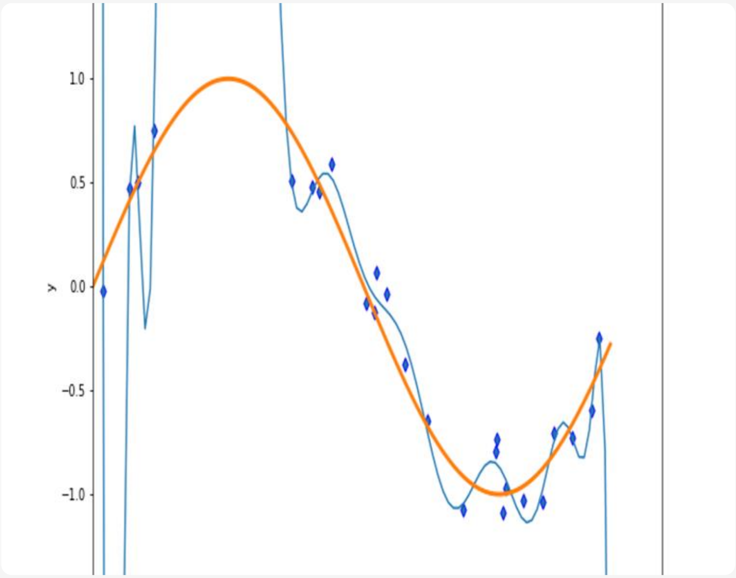
Generalization



Underfitting



Generalized



Overfitting

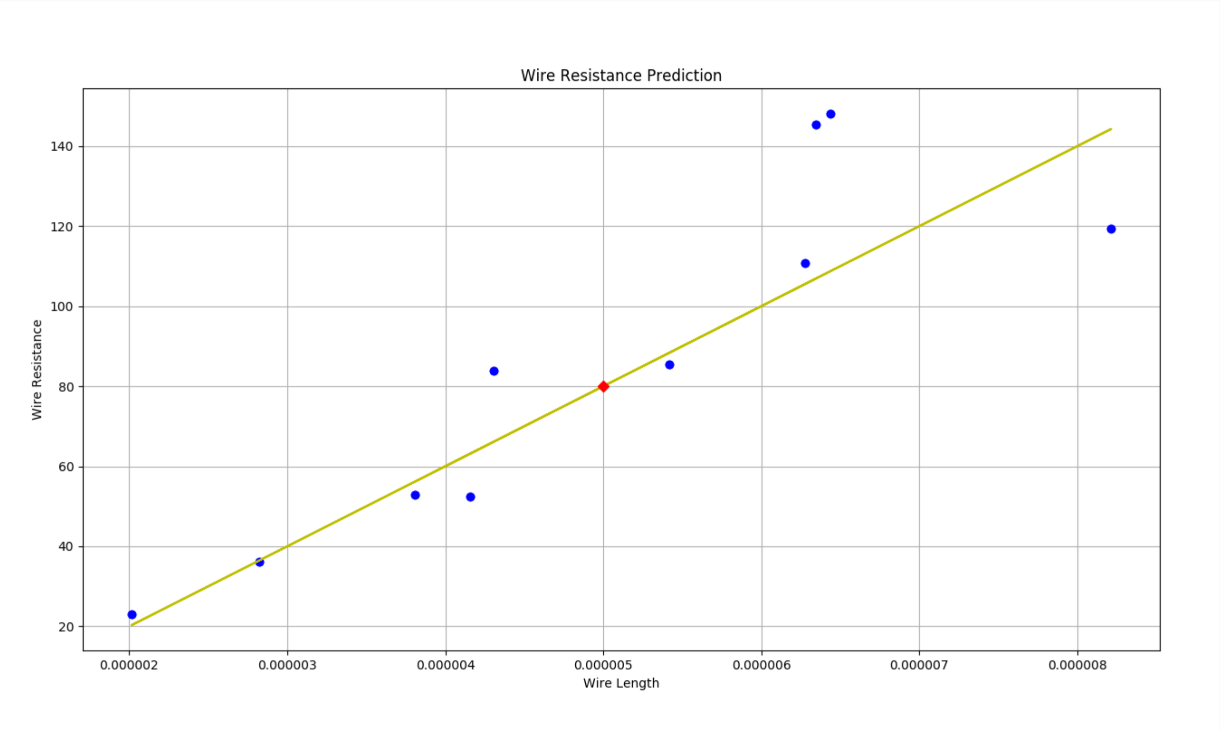
Mid Term Projects ideas

- K Nearest Neighbor (dnnc)
- K means algorithm (dnnc)
- Decision Tree (dnnc)
- Random Forests (dnnc)
- Sales Forecasting using Walmart Dataset (sklearn, dnnc)
- MNIST training (sklearn, dnnc)
- MNIST inference (dnnc)
- MNIST model compilation (dnnc)
- Anomaly Detection (dnnc)
- Your suggestions.

Regression

Regression is a form of prediction and forecasting.

Formally, it is known as a class of supervised algorithm that attempts to establish a *continuous* relationship between one dependent variable (usually shown on the Y axis) and set of other independent variables (on the X axis).



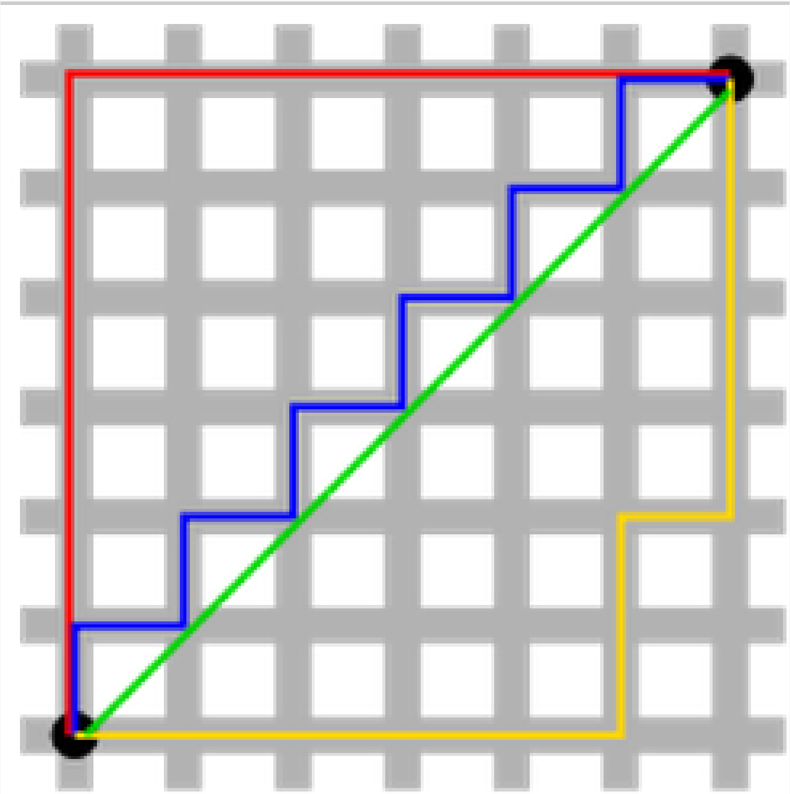
Regression and Norms

Norm is magnitude of an mathematical entity.

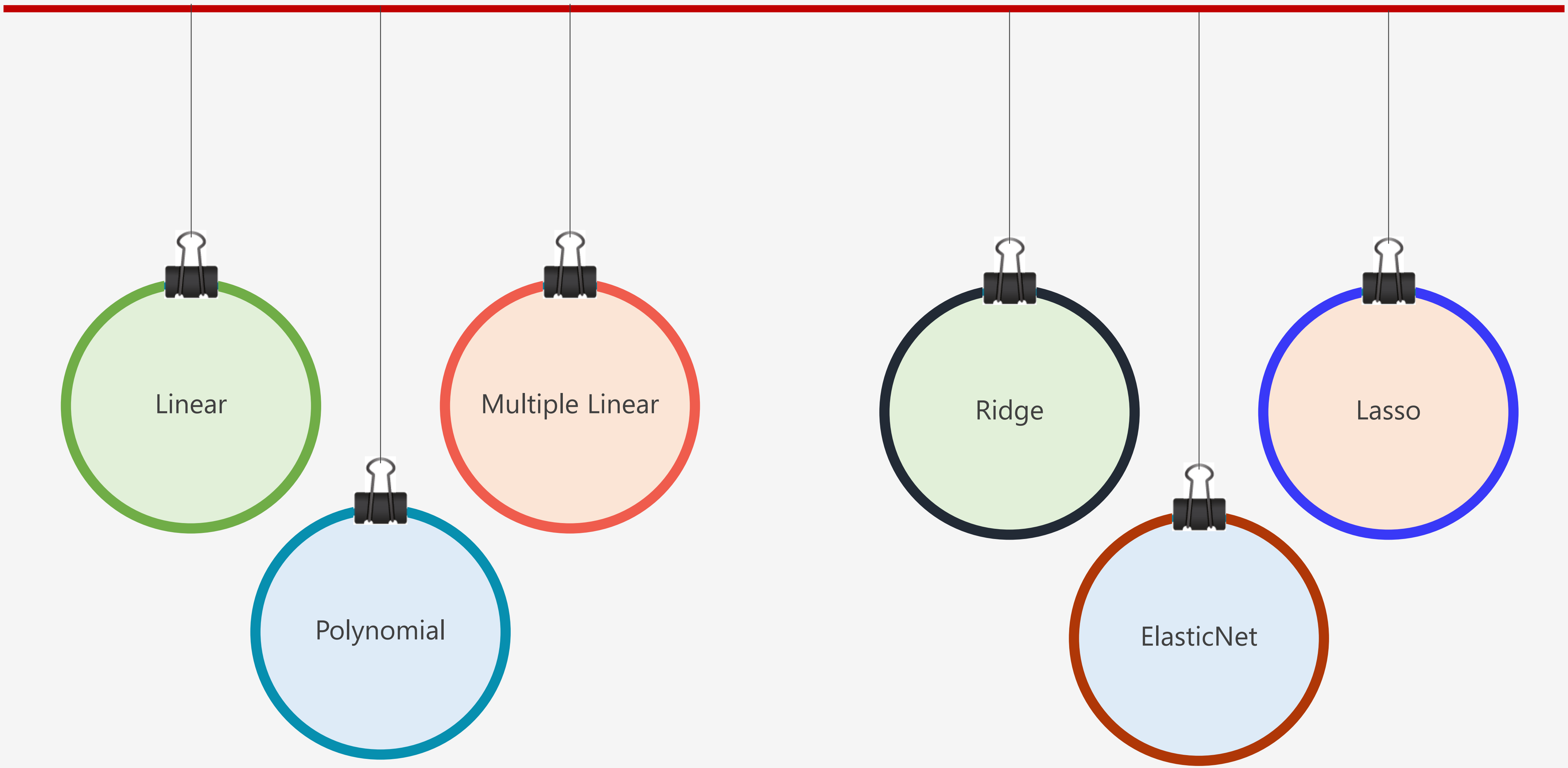
L_1 Norm = $\|x\|_1 = |x_1| + |x_2| + \ldots + |x_N|$

L_2 Norm = $\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \ldots + x_N^2}$

L_p Norm = $\|x\|_p = \sqrt[p]{x_1^p + x_2^p + \ldots + x_N^p}$



Types of regression algorithms



Types of Regression Algorithms

- Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- Ridge Regression
- Lasso Regression
- ElasticNet Regression

Linear Regression is a statistical model used to predict the relationship between independent and dependent variables denoted by x and y respectively

Examine 2 factors

1

How closely are x and y related ?

Linear regression gives a number between -1 and 1 indicating the strength of correlation between the two variables

0 : no correlation
1 : positively correlated
-1 : negatively correlated

2

Prediction

When the relationship between x and y is known, use this to predict future values of y for a value of x

This is done by fitting a regression line and represented by a linear equation:

$$y = a * x + b$$

Types of Regression Algorithms (Contd.)

- Linear Regression
- **Multiple Linear Regression**
- Polynomial Regression
- Ridge Regression
- Lasso Regression
- ElasticNet Regression

Multiple linear regression is a statistical technique used to predict the outcome of a response variable through several explanatory variables and model the relationships between them.

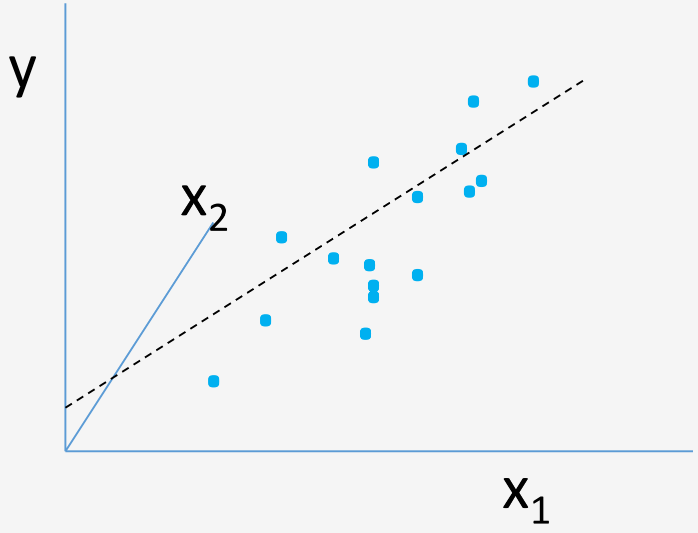
Equation for MLR

$$Y = m_1 * x_1 + m_2 * x_2 + m_3 * x_3 + \ldots\ldots\ldots + m_n * x_n + c$$

Y → Dependent Variable

m1, m2, m3 .... m_n → Slopes

c → Coefficient



Types of Regression Algorithms (Contd.)

- Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- Ridge Regression
- Lasso Regression
- ElasticNet Regression

Polynomial regression is applied when data is not formed in a straight line.

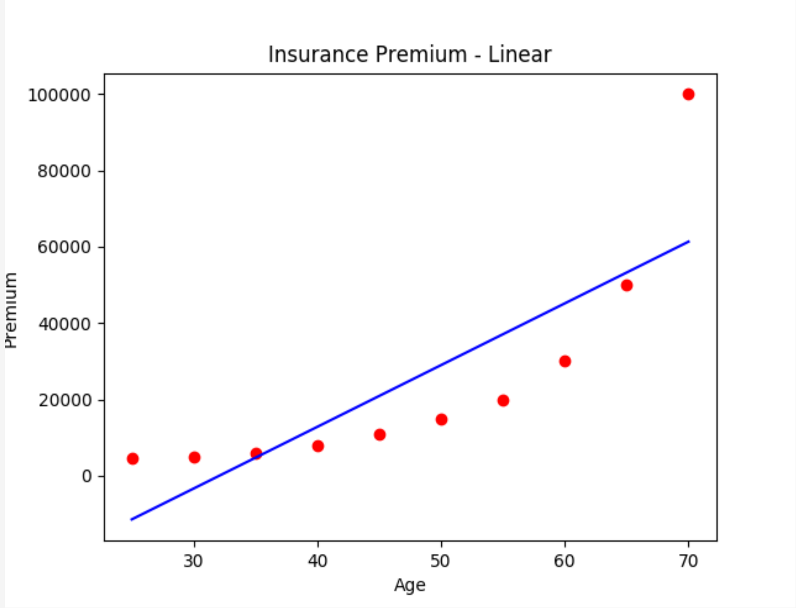
It is used to fit a linear model to non-linear data by creating new features from powers of non-linear features.

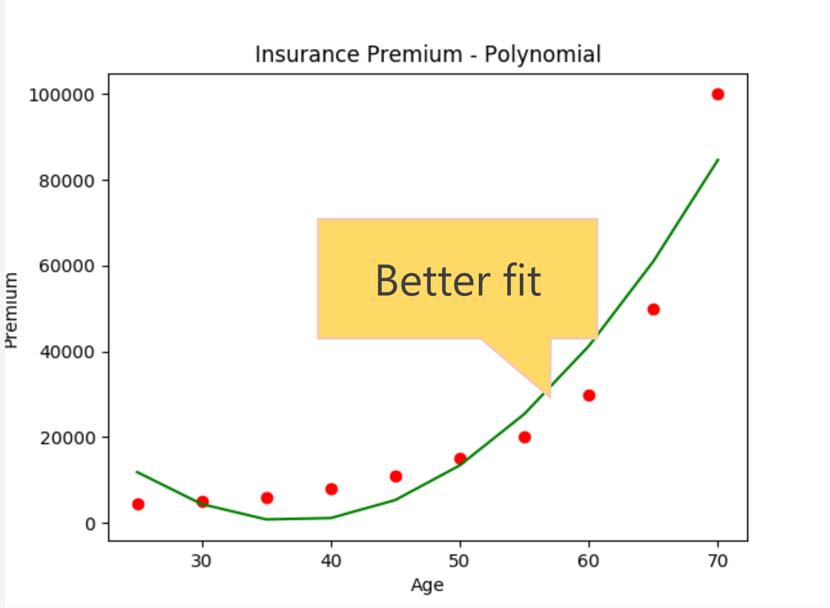
Example: Quadratic features

$$x_2' = x_2^2$$

$$y = w_1x_1 + w_2x_2^2 + 6$$

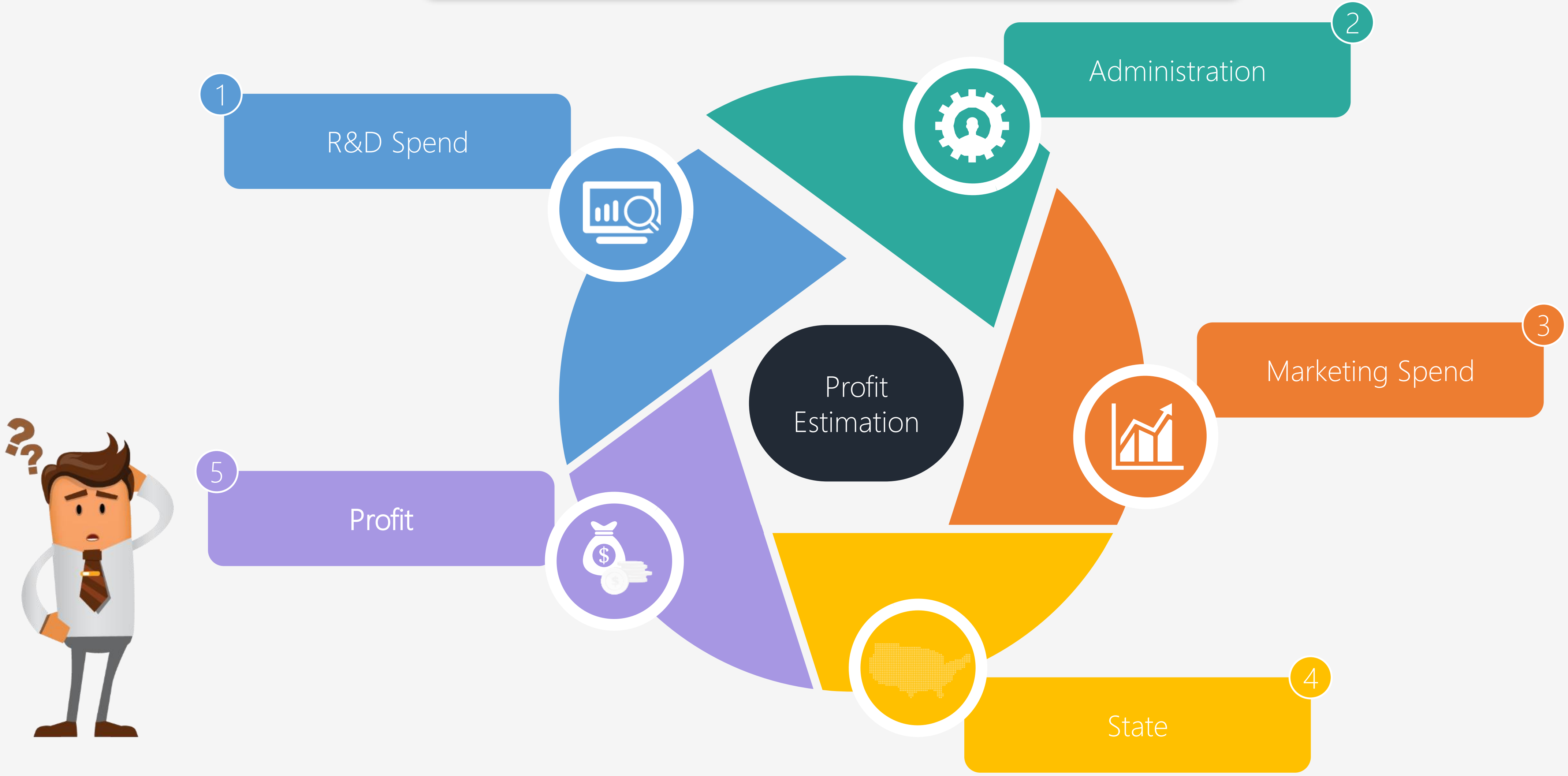
$$= w_1x_1 + w_2x_2' + 6$$



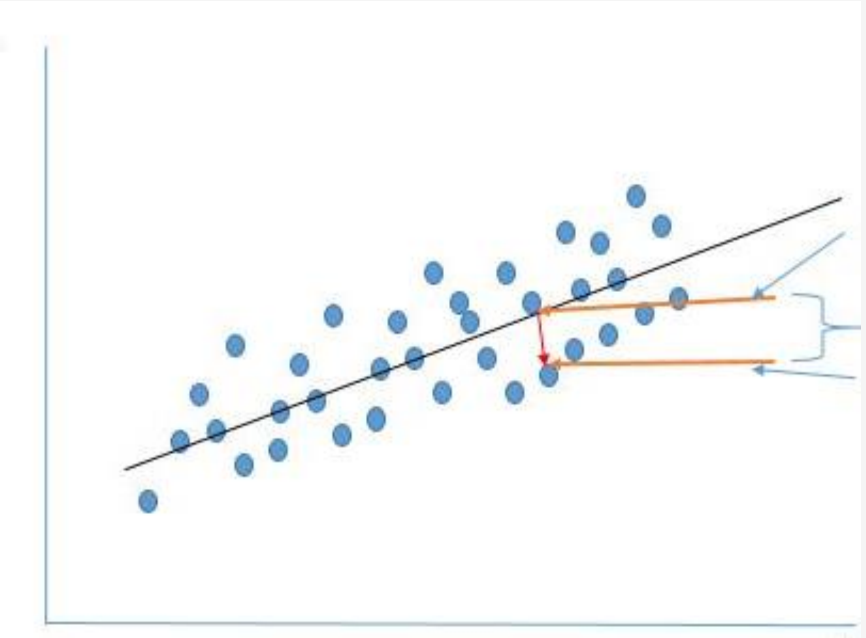


Regression Use Case

Predicting profit based on expenditures of the company



Accuracy Metrics



Predicted Value

Error

Actual Value

R-square $= 1 - \dfrac{\sum(Y\_actual - Y\_predicted)^2}{\sum(Y\_actual - Y\_mean)^2}$

- ✓ R-square is proportion of variation in the outcome that is explained by the predictor variables
- ✓ R^2 lies between 0 -100 %

Example: Performing linear regression on sq. Area (x) and Price (y) returns **R-square** value as 16. This means you have 16% information to make an accurate prediction about the price.

Adjusted R-Squared

The disadvantage with R-squared is that it assumes every independent variable in the model explains variations in the dependent variable.

Use adjusted R-squared when working on a multiple linear regression problem.

$$Adjusted\ R^2 = 1 - \frac{(1 - R^2)(N - 1)}{N - p - 1}$$

where R^2 is R-squared value

P is number of predictor variables

N is number data points

Adjusted R_2 is useful in dataset with many variables in the model.

Cost Function

Mean-Squared Error (MSE) is also used to measure the performance of a model.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y}_i)^2$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y}_i)^2}$$

Where *N* is the number of data points

y_i is the predicted value by the model

$\bar{y}_i$ is the actual value for the data point

These functions are called the loss function or the cost function, and the value has to be minimized.

Evaluating Coefficients

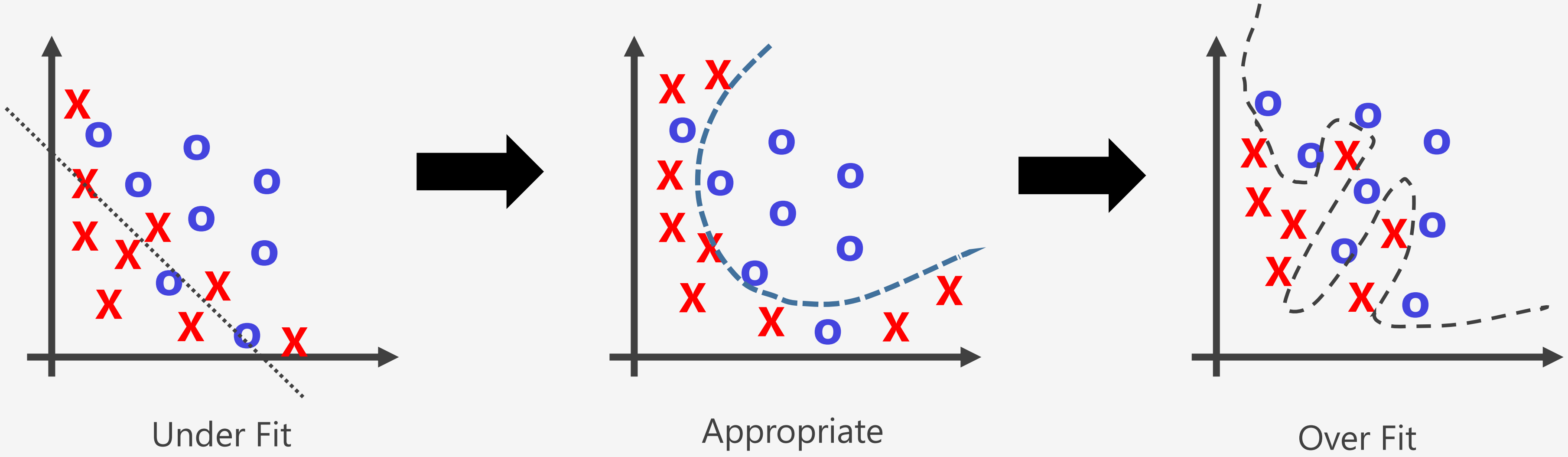
In regression analysis, p-values and coefficients together indicate which relationships in the model are statistically significant and the nature of those relationships.

Coefficients describe the mathematical relationship between each independent variable and the dependent variable.

p-values for the coefficients indicate whether these relationships are statistically significant.

$p < 0.05$	**REJECT** the Null hypothesis, meaning variables have some effect and need to be retained
$p > 0.05$	**ACCEPT** the Null hypothesis, meaning variables have no effect and can be removed

Challenges in Prediction



If the model learning is poor, you have an **underfitted** situation

The algorithm will not work well on test data
Retraining may be needed to find a better fit

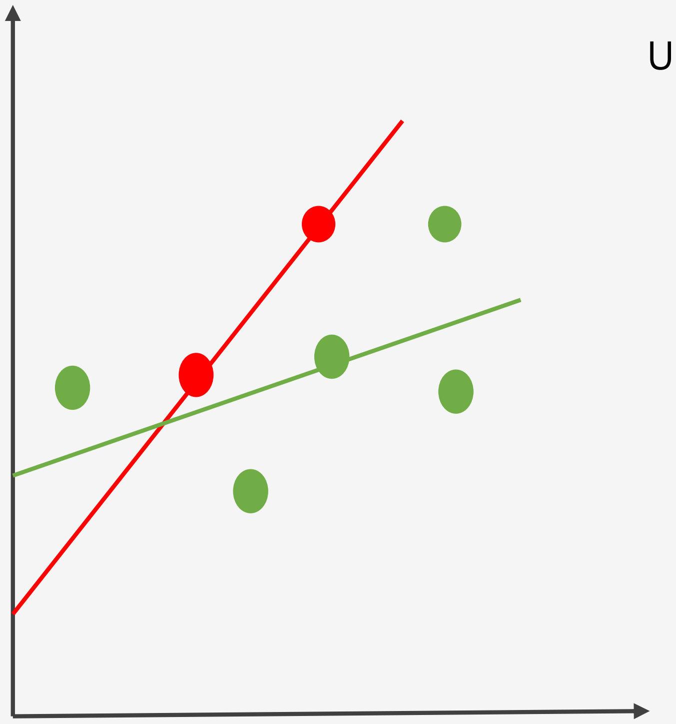
Overfitting happens when model accuracy for training data is good, but model does not generalize well to the overall population

Algorithm is not able to give good predictions for the new data

Regularization

Regularization solves overfitting to the training data.

Used to restrict the parameters values that are estimated in the model



Training

Testing

$$\boxed{L = \sum(\hat{Y}_i - Y_i)^2 + \alpha \sum \beta^2}$$

This loss function includes 2 elements.

1) the sum of square distances between predicted and actual value

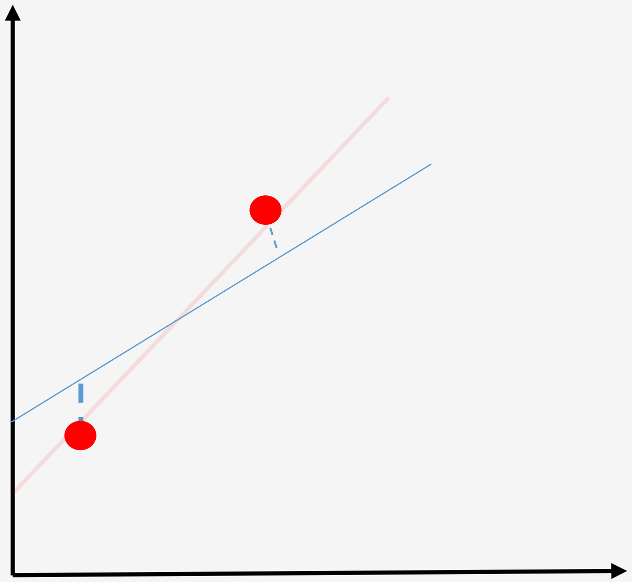
2) the second element is the regularization term

Types of Regression (Contd.)

- Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- **Ridge Regression**
- Lasso Regression
- ElasticNet Regression

Ridge Regression (L2) is used when there is a problem of multicollinearity, By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors.

The main idea is to find a new line that has some bias with respect to the training data
In return for that small amount of bias, a significant drop in variance is achieved



Minimization objective = LS Obj + α * (sum of the square of coefficients)

LS Obj refers to least squares objective

α controls the strength of the penalty term

Types of Regression (Contd.)

- Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- Ridge Regression
- **Lasso Regression**
- ElasticNet Regression

Lasso Regression (L1) is similar to ridge, but it also performs feature selection.

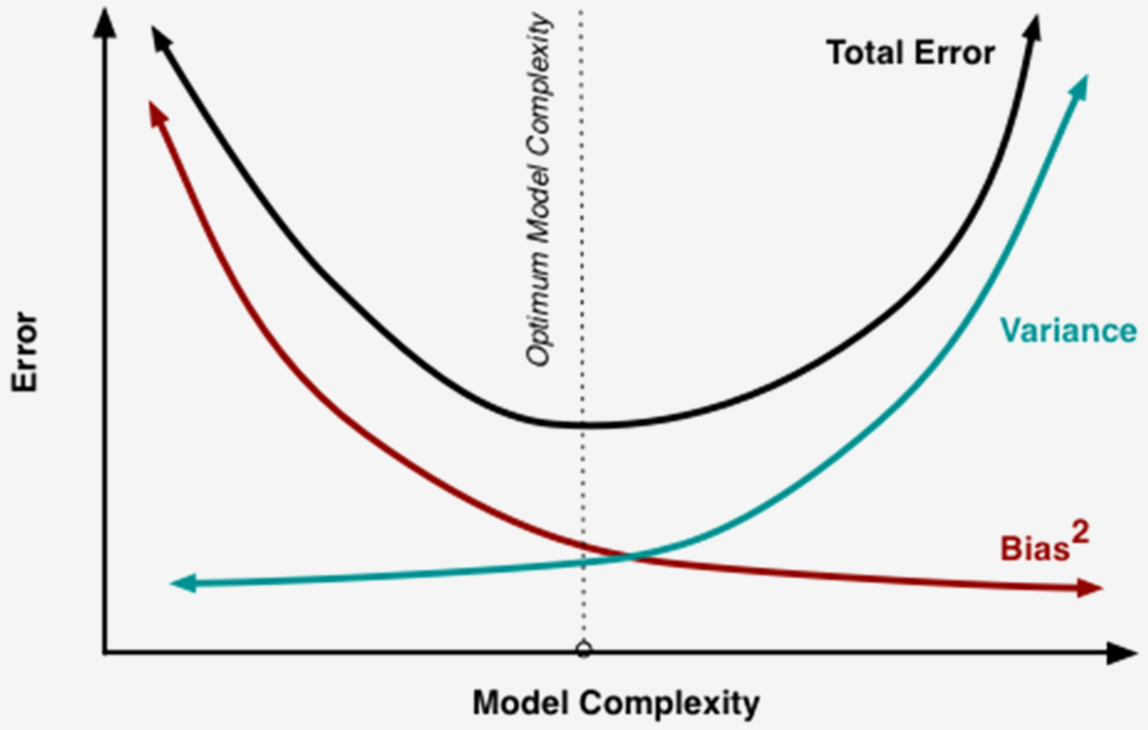
It will set the coefficient value for features that do not help in decision making very low, potentially zero.

Minimization objective = LS Obj + α * (sum of absolute coefficient values)

Lasso regression tends to exclude variables that are not required from the equation, whereas ridge tends to do better when all variables are present.

Bias vs Variance

- bias is related with a model failing to fit the training set and variance is related with a model failing to fit the testing set.
- Bias and variance are in a trade-off relationship over model complexity, which means that
  - a simple model would have high-bias and low-variance, and
  - vice versa.



Types of Regression (Contd.)

- Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- Ridge Regression
- Lasso Regression
- ElasticNet Regression

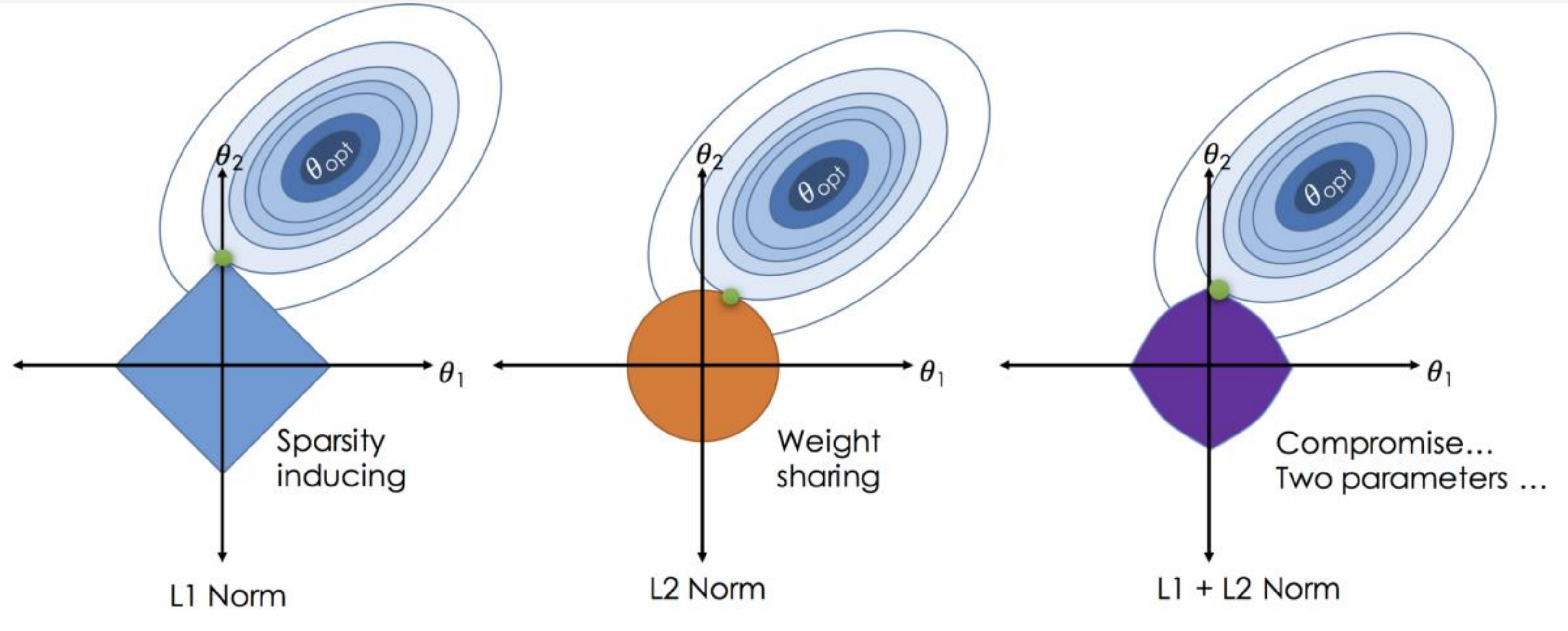
ElasticNet regression combines
the strength of **lasso and ridge regression**

$$\boxed{\begin{gathered}\textbf{the sum of the squared residuals} \\ + \\ \underbrace{\lambda_1 \times |\text{variable}_1| + \dots + |\text{variable}_x|}_{\text{Lasso penalty}} + \underbrace{\lambda_2 \times \text{variable}_1^2 + \dots + \text{variable}_x^2}_{\text{Ridge penalty}}\end{gathered}}$$

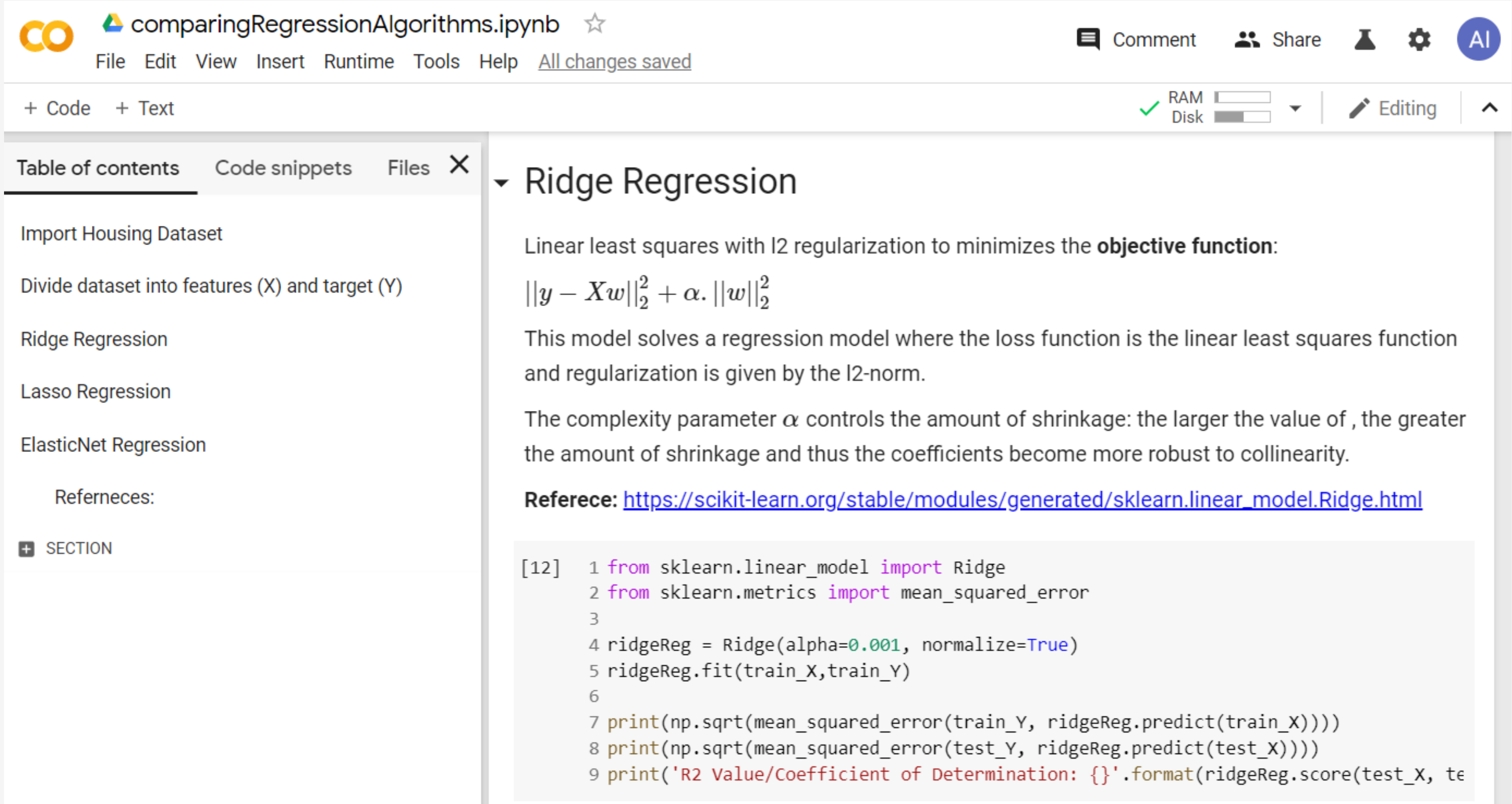
If you are not sure whether to use lasso or ridge, use ElasticNet

Regression Practice

- **Problem Statement:** Compare and contrast Ridge, Lasso and ElasticNet regression for the California House Pricing dataset used earlier.
- **Objective:**
  - Build a predictive model using Ridge, Lasso and ElasticNet



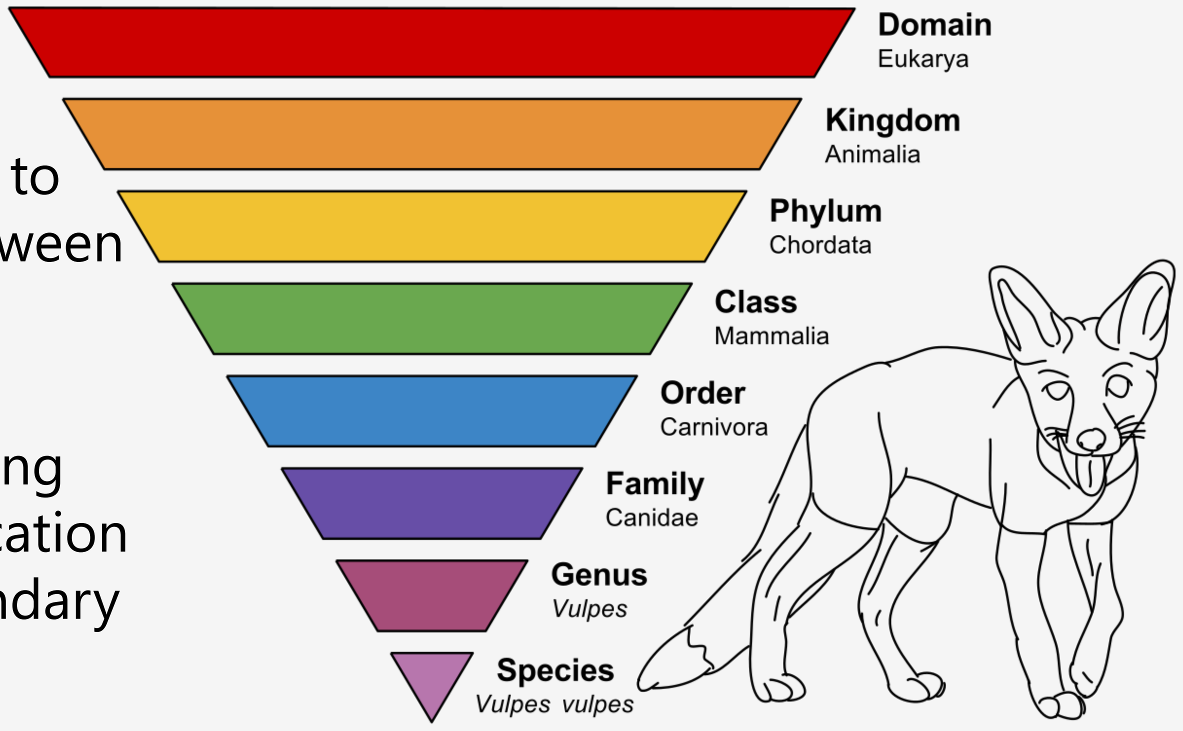
Answer



https://colab.research.google.com/drive/1_W-L7cP3Ji8rLXi8fOr3nxB9wrcjNQXP

Classification

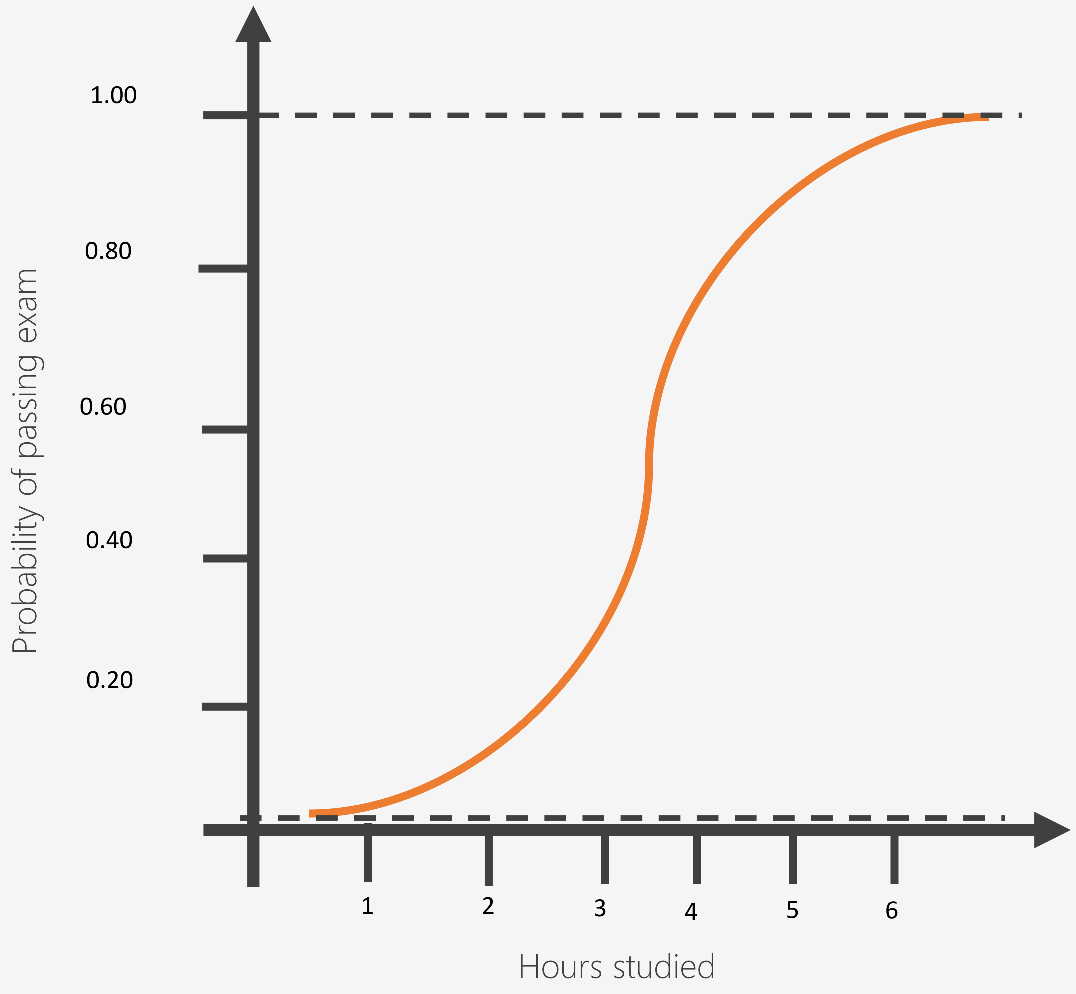
- Classification is a process of arranging objects into taxonomic groups according to observed qualities or characteristics.
- In machine learning paradigm, classification is an instance of a supervised algorithm that attempts to establish a discrete relationship between dependent variable and set of independent variables.
- one can convert any machine learning problem from regression to classification with a concept called decision boundary



Red fox (*Vulpes vulpes*)

Logistic Regression

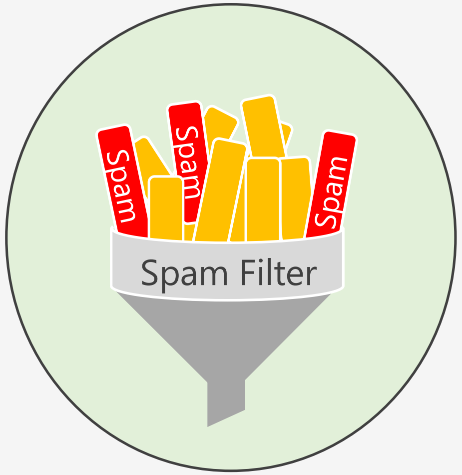
- Logistic Regression is widely used to predict binary outcomes for a given set of independent variables.
- The dependent variable's outcome is discrete such as $y \in \{0, 1\}$
- A binary dependent variable can have only two values such as 0 or 1, win or lose, pass or fail, healthy or sick.



Use Cases

Loan sanction

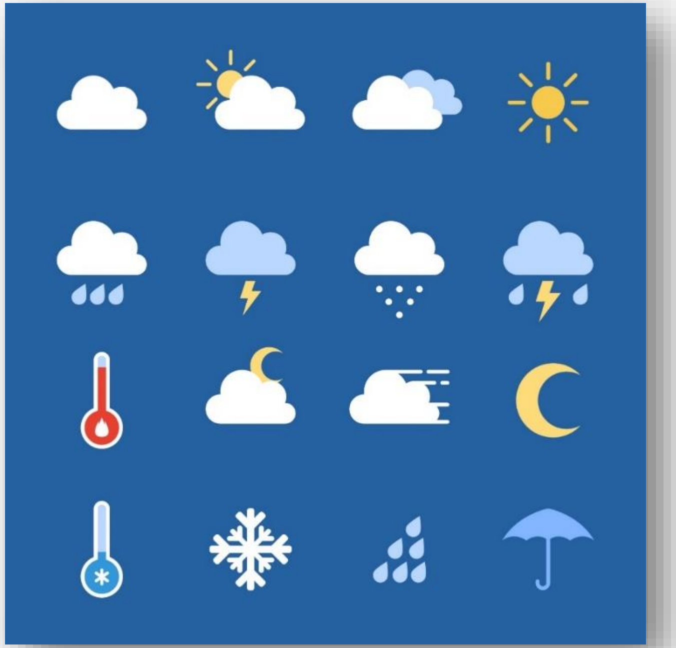
Customer segments



Spam filtering

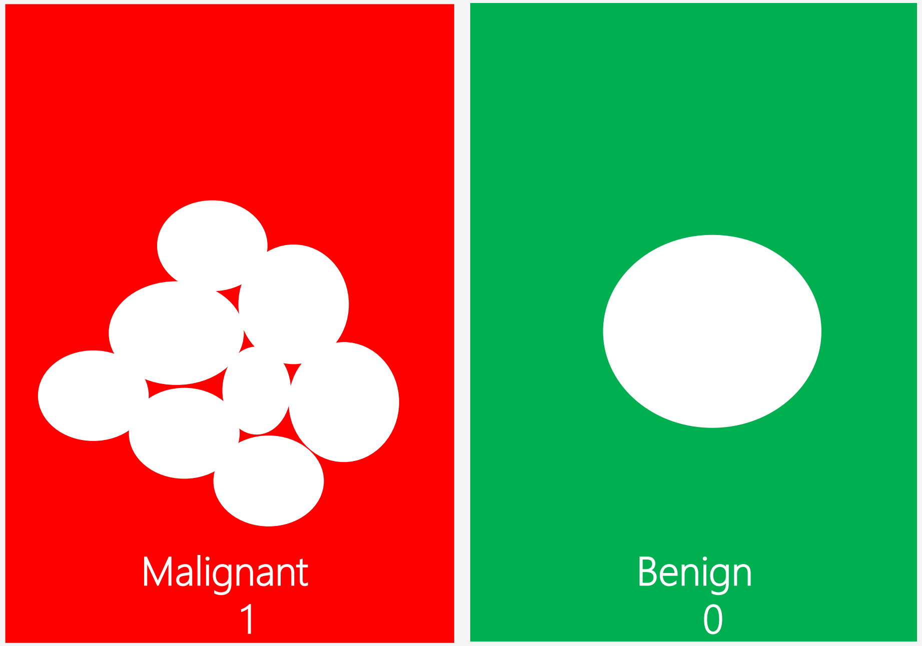
Exam results – Pass/Fail

Real-Life Scenarios



Weather Forecast

sunny, stormy, cloudy, rainy



Cancer Prediction

Malignant (cancerous) and Benign (non-cancerous)

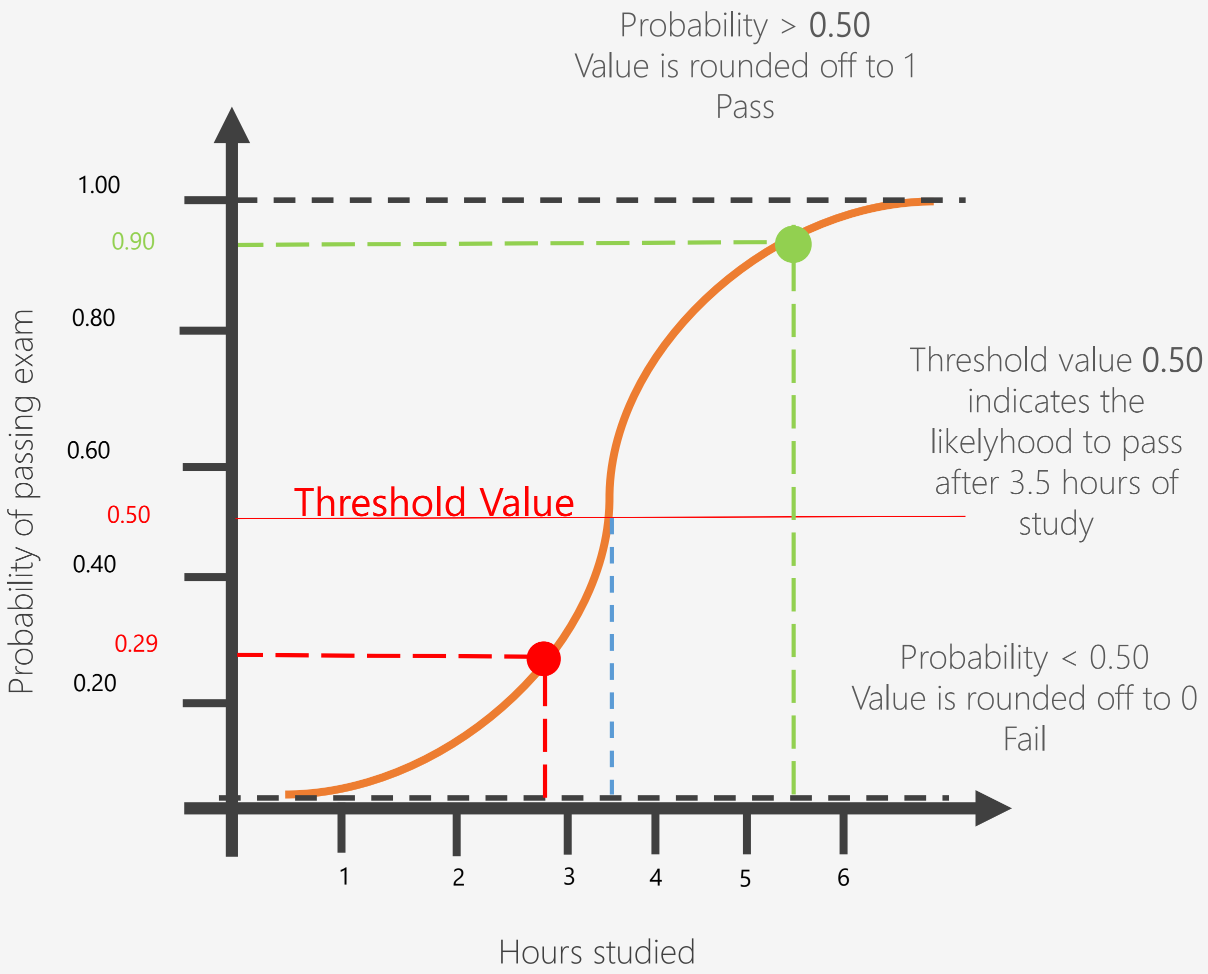
Logistic Regression (Contd.)

The probability distribution of output y is restricted to 1 or 0.
This is called as **sigmoid probability (σ)**

If $\sigma > 0.5$, set $y = 1$, else set $y = 0$.

Unlike Linear Regression (and its Normal Equation solution), there is no closed form solution for finding optimal weights of Logistic Regression.

Instead, you must solve this with **maximum likelihood estimation** (probability model to detect maximum likelihood of something happening).



Logistic Regression Equation

The Logistic regression equation is derived from the straight line equation:

Equation of a straight line

$Y = w.x+b$ → Range is from – (infinity) to (infinity)

Deducing the logistic regression equation from straight line equation

$Y = w.x+b$ → In logistic equation, Y can be only from 0 to 1

Transform it to get the range

$$\frac{Y}{1-Y} \left\{ \begin{array}{l} Y = 0 \text{ then } 0 \\ Y = 1 \text{ then infinity} \end{array} \right.$$

→ Now, the range is between 0 to infinity

Transform it further to get range: (infinity) to (infinity)

$$\boxed{\log\left[\frac{Y}{1-Y}\right] \Rightarrow w.x+b}$$

→ Final Logistic Regression Equation

Sigmoid Probability

The probability in the logistic regression is represented by the Sigmoid function (logistic function or the S-curve).

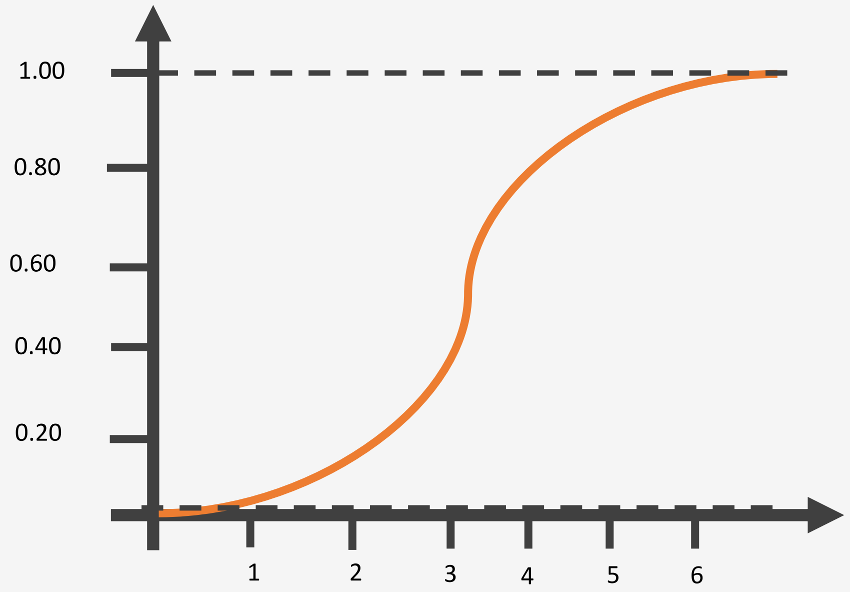
$$S(t) = \frac{1}{1 + e^{-t}}$$

t represents data values * number of hours studied
S(t) represents the probability of passing the exam.

The sigmoid function gives an 'S' shaped curve.

This curve has a finite limit that is Y can only be 0 or 1

0 as x approaches to $-\infty$
1 as x approaches to $+\infty$



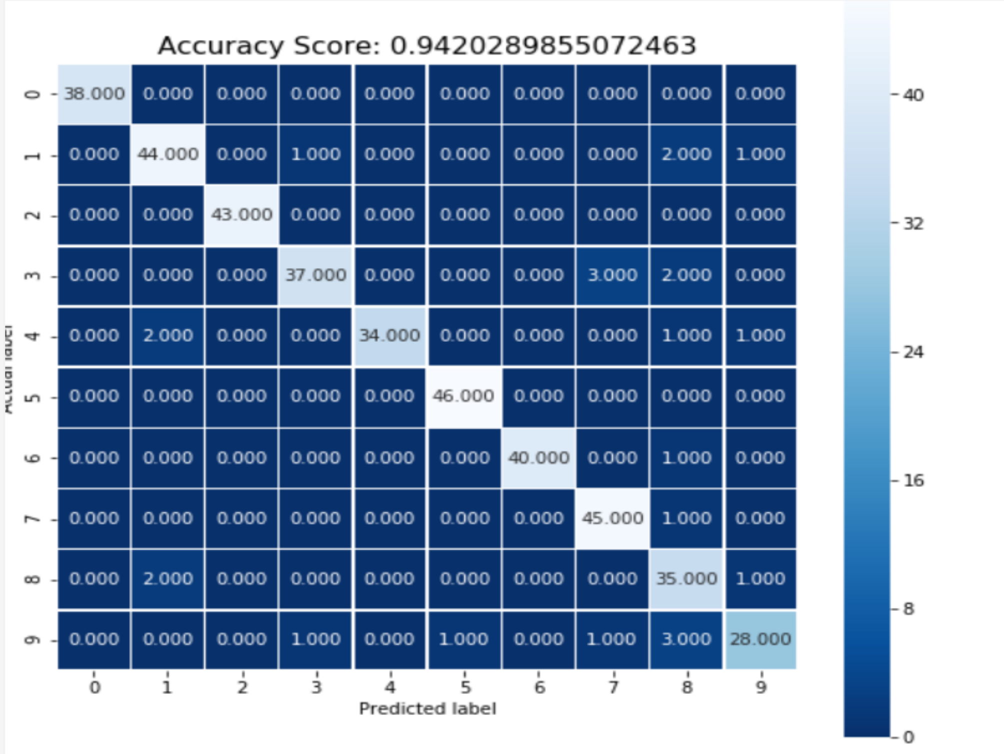
Accuracy Metrics

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

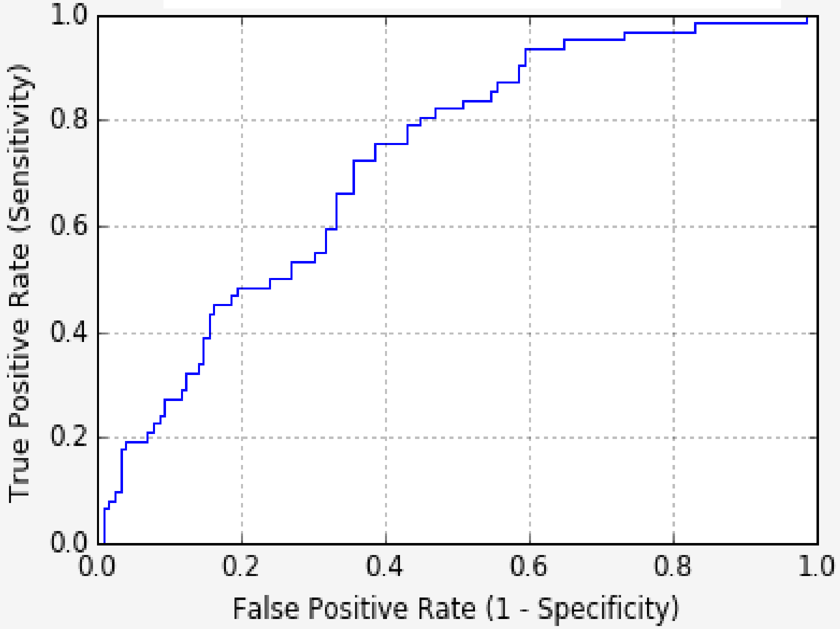
Confusion Matrix

$$\textbf{Accuracy} = \frac{\text{True Positive (TP)} + \text{True Negative (TN)}}{\text{Total}}$$

Confusion Matrix for a multi - class classification

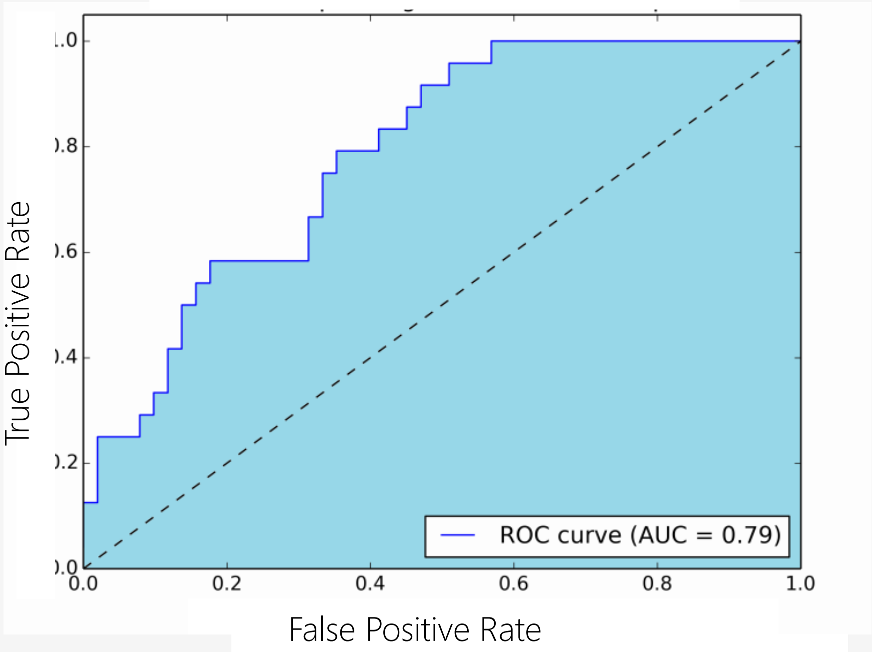


Accuracy Metrics (Contd.)



ROC curve

Compares the model true positive and false positive rates to the ones from a random assignment

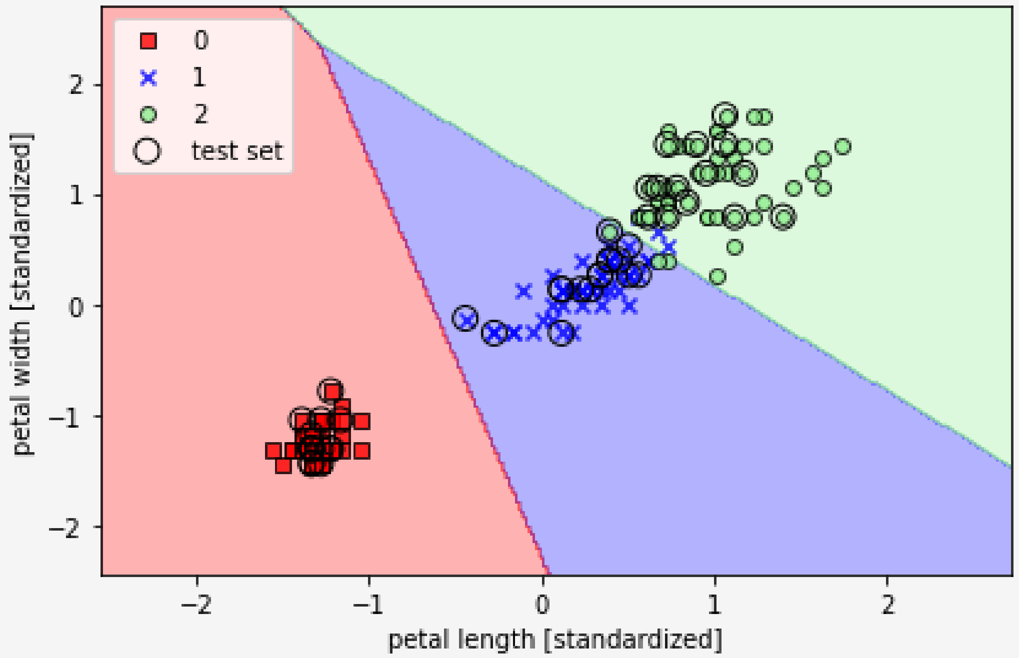


AUC (Area under the ROC Curve)

Measures the entire two-dimensional area under the entire ROC curve

Classification Practice

- **Problem Statement:** The Iris plant has 3 species - Iris Setosa, Iris Versicolour, Iris Virginica. One class is linearly separable from the other two; the latter are not linearly separable from each other.
- **Objective:**
  - Import the iris dataset using sklearn
  - Use logistic regression to predict the class of iris plant



Answer

Code

Importing the required libraries

```
from sklearn import datasets
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score

import matplotlib.pyplot as plt
import matplotlib.colors
from sklearn.linear_model import LogisticRegression
```

Importing the dataset

```
iris = datasets.load_iris()
X = iris.data[:, [2, 3]]
y = iris.target

print('Class labels:', np.unique(y))
```

Answer (Contd.)

Code

Train test split

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=1, stratify=y)

print('Labels counts in y:', np.bincount(y))
print('Labels counts in y_train:', np.bincount(y_train))
print('Labels counts in y_test:', np.bincount(y_test))
```

Standardizing features

```
sc = StandardScaler()
sc.fit(X_train)
X_train_std = sc.transform(X_train)
X_test_std = sc.transform(X_test)
```

Answer (Contd.)

Code

Plotting decision surface

```
def plot_decision_regions(X, y, classifier, test_idx=None, resolution=0.02):
    # setup marker generator and color map
    markers = ('s', 'x', 'o', '^', 'v')
    colors = ('red', 'blue', 'lightgreen', 'gray', 'cyan')
    cmap = matplotlib.colors.ListedColormap(colors[:len(np.unique(y))])

    # plot the decision surface
    x1_min, x1_max = X[:, 0].min() - 1, X[:, 0].max() + 1
    x2_min, x2_max = X[:, 1].min() - 1, X[:, 1].max() + 1
    xx1, xx2 = np.meshgrid(np.arange(x1_min, x1_max, resolution),np.arange(x2_min, x2_max,
resolution))
    Z = classifier.predict(np.array([xx1.ravel(), xx2.ravel()]).T)
    Z = Z.reshape(xx1.shape)
    plt.contourf(xx1, xx2, Z, alpha=0.3, cmap=cmap)
    plt.xlim(xx1.min(), xx1.max())
    plt.ylim(xx2.min(), xx2.max())
```

Answer (Contd.)

Code

Plotting decision surface

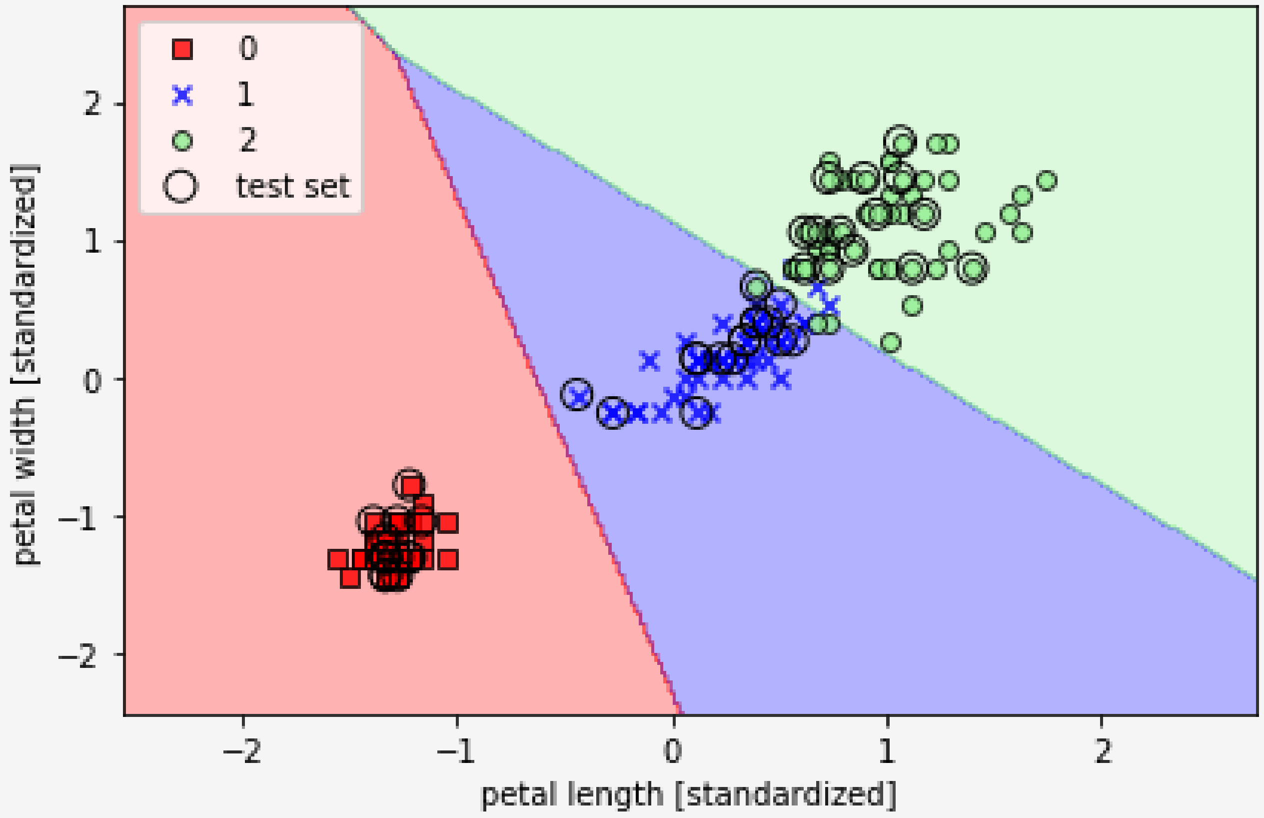
```
for idx, cl in enumerate(np.unique(y)):
        plt.scatter(x=X[y == cl, 0], y=X[y == cl, 1],alpha=0.8,
c=colors[idx],marker=markers[idx], label=cl, edgecolor='black')

    if test_idx:
        X_test, y_test = X[test_idx, :], y[test_idx]
        plt.scatter(X_test[:, 0], X_test[:,
1],c='',edgecolor='black',alpha=1.0,linewidth=1,marker='o',s=100, label='test set')

X_combined_std = np.vstack((X_train_std, X_test_std))
y_combined = np.hstack((y_train, y_test))
```

Answer (Contd.)

Output plot



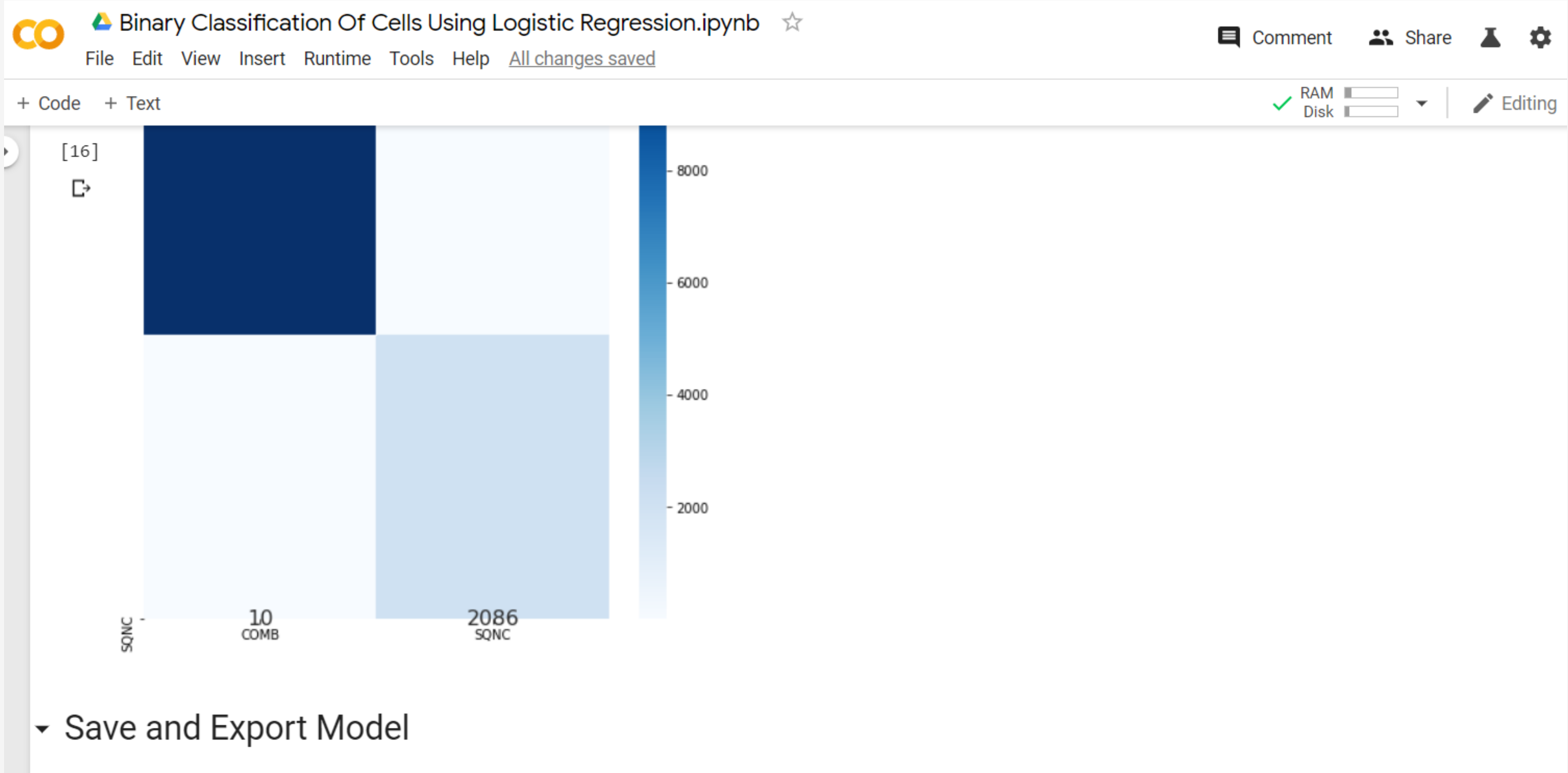
Answer (Contd.)

Code

Training logistic regression model

```
lr = LogisticRegression(C=100.0, random_state=1)
lr.fit(X_train_std, y_train)
plot_decision_regions(X_combined_std, y_combined,
                      classifier=lr, test_idx=range(105, 150))
plt.xlabel('petal length [standardized]')
plt.ylabel('petal width [standardized]')
plt.legend(loc='upper left')
plt.tight_layout()
#plt.savefig('images/03_06.png', dpi=300)
plt.show()
lr.predict_proba(X_test_std[:3, :])
lr.predict_proba(X_test_std[:3, :]).sum(axis=1)
lr.predict_proba(X_test_std[:3, :]).argmax(axis=1)
lr.predict(X_test_std[:3, :])
lr.predict(X_test_std[0, :].reshape(1, -1))
```

Logistic Regression on Cells

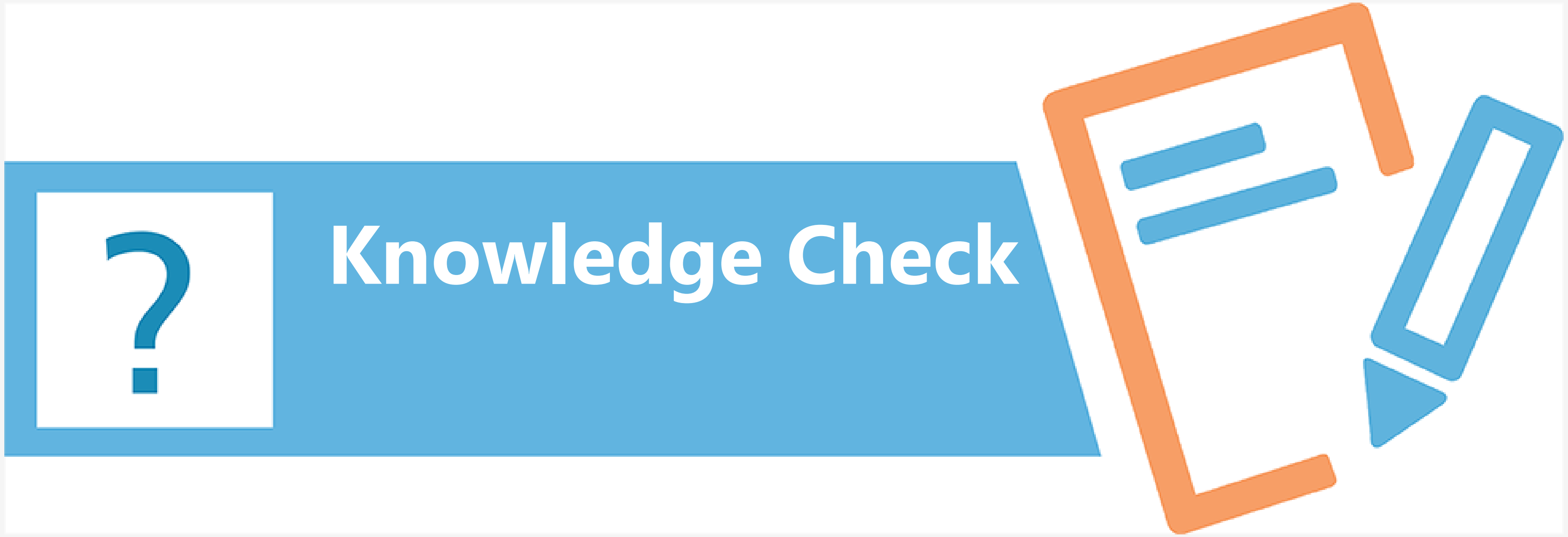


https://colab.research.google.com/drive/18m9iwOwX6Q7NtCGqanAbbhX8ZlTwdSar

Key Takeaways

Now, you are able to:

- Understand the different types of supervised learning
- Build various regression models


?
Knowledge Check

Knowledge Check

1

In the equation of a straight line $Y = mX + c$, the term m is the:

a. Slope

b. Independent Variable

c. Dependent Variable

d. Intercept

Knowledge Check

1

In the equation of a straight line Y = mX + c, the term m is the:

a. Slope

b. Independent Variable

c. Dependent Variable

d. Intercept

The correct answer is **a. Slope**

In the equation of a straight line Y = mX + c, m represents the slope, and c is any constant.

Knowledge Check 2

The standard error of the estimate is a measure of:

a. Explained variation

b. Variation around the regression line

c. Variation of the X variable

d. Total variation of the Y variable

Knowledge Check

2

The standard error of the estimate is a measure of:

a. Explained variation

b. Variation around the regression line

c. Variation of the X variable

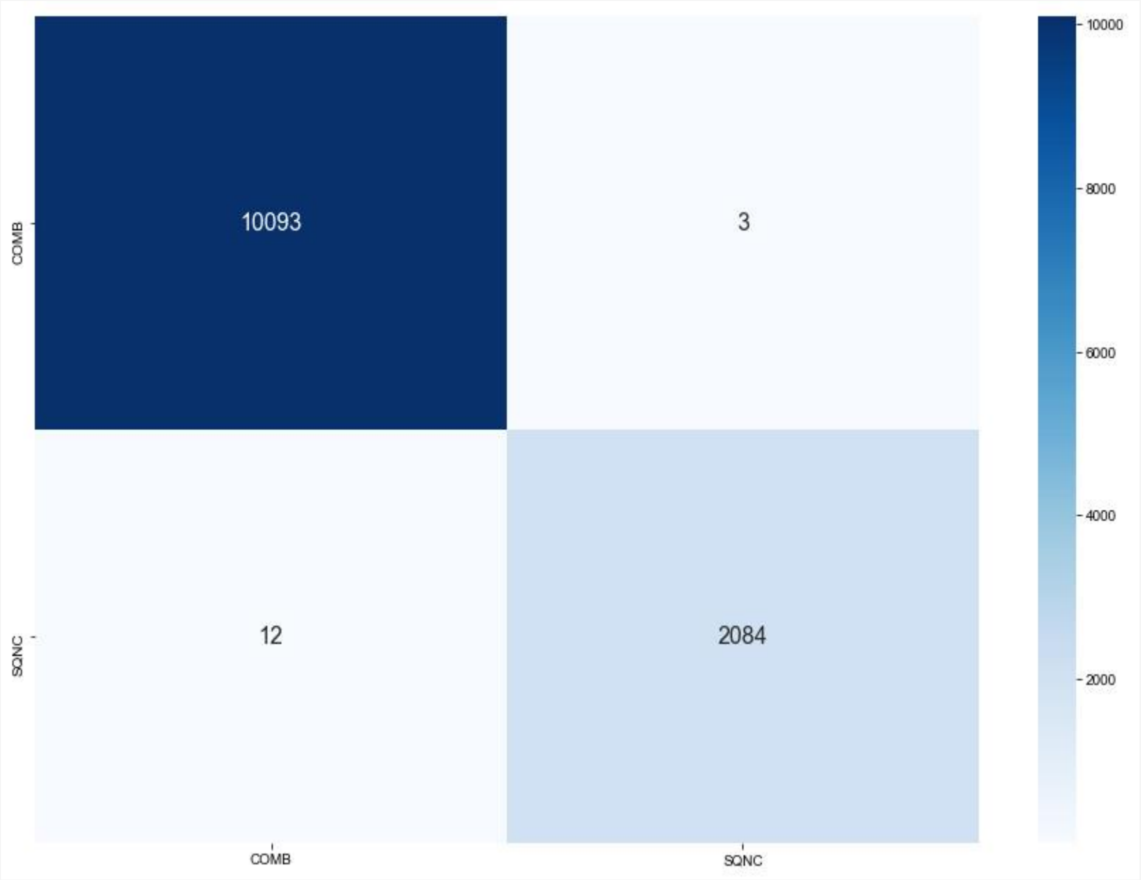
d. Total variation of the Y variable

The correct answer is **b. Variation around the regression line**

The standard error of the estimate is a measure of the variation around the regression line.

Accuracy

- Regression accuracy is measured using cumulative error, low error means better model
- Classification accuracy involves confusion matrix to capture F-Score
  - Precision
  - Recall
  - Accuracy
  - Error Rate



AI-first advocate

hi@ai-techsystems.com

AI Tech Systems

Putting AI-First

105 Serra Way,
#251
Milpitas, CA, 95035

+1-408-372-7405

www.ai-techsystems.com

California Science and Technology UNIVERSITY