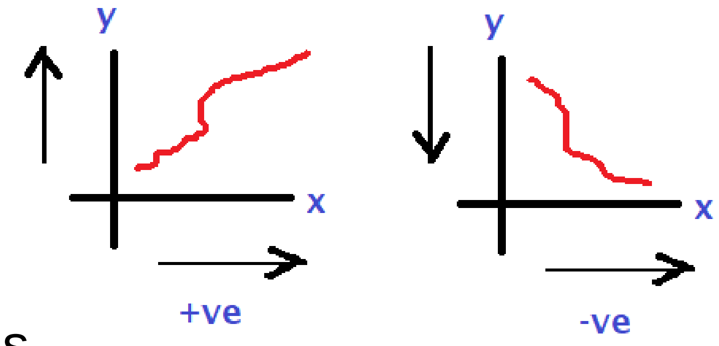
MACHINE LEARNING
End to End Process

Trainer : Sujata Mohite
Email: sujata.mohite@sunbeaminfo.com

Measures of Correlation

Terminology

- **Univariate**
  - This type of data consists of only one variable (eg. Temperature value)
  - It does not deal with causes or relationships
  - the main purpose of the analysis is to describe the data and find patterns that exist within it
- **Bivariate**
  - This type of data involves two different variables
  - The analysis of this type of data deals with causes and relationships
  - the analysis is done to find out the relationship among the two variables

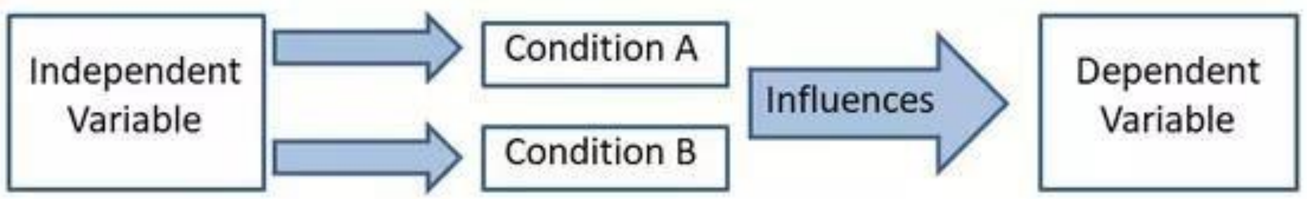


- **Multivariate**
  - When the data involves three or more variables
  - It is similar to bivariate but contains more than one independent variable

Terminology

- **Independent variable(s)**
  - A variable that represents a quantity that is being manipulated in an experiment
  - **Represents input**
  - Also known as regressors in a statistical context.
  - x is often the variable used to represent the independent variable in an equation

- **Dependent variable**
  - A quantity whose value *depends* on how the independent variable is manipulated
  - **Represents output**
  - y is often the variable used to represent the independent variable in an equation

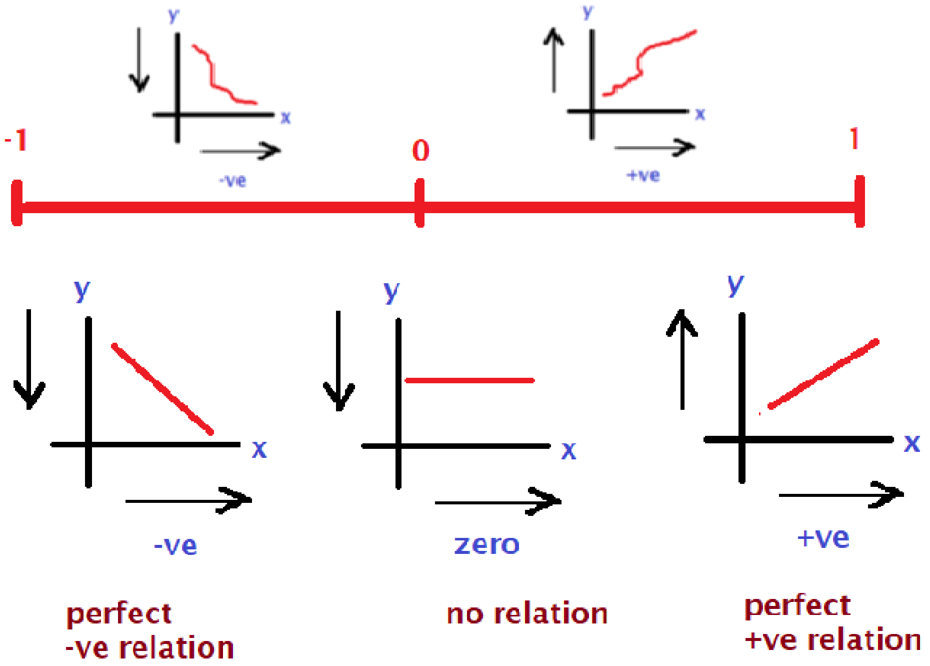


Correlation

- Measures the strength of the relationship between two or more variables
- Correlation is the scaled measure of covariance
- It is dimensionless: the correlation coefficient is always a pure value and not measured in any units

$$\rho(X,Y) = \frac{cov(X,Y)}{\sigma_x \sigma_y}$$

- Where
  - ρ(X,Y) – the correlation between the variables X and Y
  - cov(X,Y) – the covariance between the variables X and Y
  - σ_X – the standard deviation of the X-variable
  - σ_Y – the standard deviation of the Y-variable



Correlation

- Even a high degree of correlation does not necessarily mean that the relationship of cause and effect exists between the variables.

- The explanation of significant degree of correlation may be one or both of the following-
    - The correlation may be due to pure chance, especially in a small sample-
    - Both the correlated variables may be influenced by one or more other variables
    - Both the correlated variables are mutually influencing each other.

Correlation coefficient

- Following are the ways to calculate correlation coefficient
  - Karl Pearson’s coefficient of correlation
  - Spearman’s Rank correlation
  - Scatter diagram
  - Coefficient of concurrent duration

- Correlation (r) → corr(x , y) = corr(y , x)
  - The coefficient of correlation lies between -1 and +1, symbolically -1 <= r <= 1
  - r = 1 (perfect correlation)
  - r = -1 (perfect negative correlation)
  - r > 0 (positive correlation)
  - r < 0 (negative correlation)
  - r = 0 (no correlation)

Covariance vs Correlation

- Both primarily assess the relationship between variables
- The closest analogy to the relationship between them is the relationship between the variance and standard deviation
- Covariance measures the total variation of two random variables from their expected values while correlation measures the strength of the relationship between variables
- Using covariance, we can only gauge the direction of the relationship while correlation is the scaled measure of covariance

End to End Process

Steps

- Look at the big picture
- Get the data → collect/organize data
- Discover and visualize the data to gain insights (**Exploratory Data Analysis** (**EDA**))
- Prepare the data for Machine Learning algorithms → data cleansing
- Select a model and train it → by trial and error method
- Fine-tune your model
- Present your solution
- Launch, monitor, and maintain your system

Look at the Big Picture

- Frame the Problem(Domain Knowledge)
    - The first question to ask your boss is what exactly the business objective is Building a model is probably not the end goal
    - How does the company expect to use and benefit from this model?
    - Knowing the objective is important because it will determine
        - how you frame the problem
        - which algorithm you will select
        - which performance measure you will use to evaluate your model
        - how much effort you will spend tweaking it

- Select a Performance Measure
    - Your next step is to select a performance measure
    - A typical performance measure for **regression** problems is the **Root Mean Square Error (RMSE) and MAE(Mean Absolute Error)**
    - It gives an idea of how much error the system typically makes in its predictions, with a higher weight for large errors

Performance Measures

Regression

- RMSE (Root Mean Squared Error) :square root of Mean Squared error
- MAE (Mean Absolute Error) : the average of the absolute difference between the actual and predicted values in the dataset
- MSE :the average of the squared difference between the original and predicted values in the data set

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

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}|$$

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

Where,

$\hat{y}$ – *predicted value of y*

$\bar{y}$ – *mean value of y*

The lower value of MAE, MSE, and RMSE implies higher accuracy of a regression model

Performance Measures

Classification

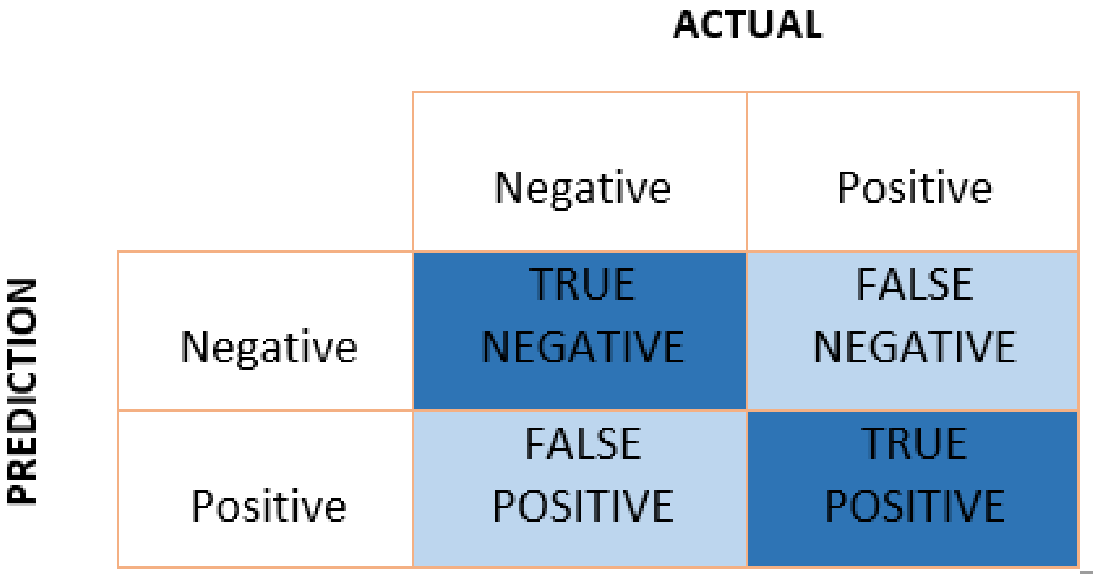
- Accuracy Score : measures how often the classifier correctly predicts
- F1-Score : the harmonic mean of precision and recall.
- Confusion Matrix :It is a matrix of size 2×2 for binary classification with actual values on one axis and predicted on another

$$\mathbf{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$F1 = 2.\frac{Precision \times Recall}{Precision + Recall}$$

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

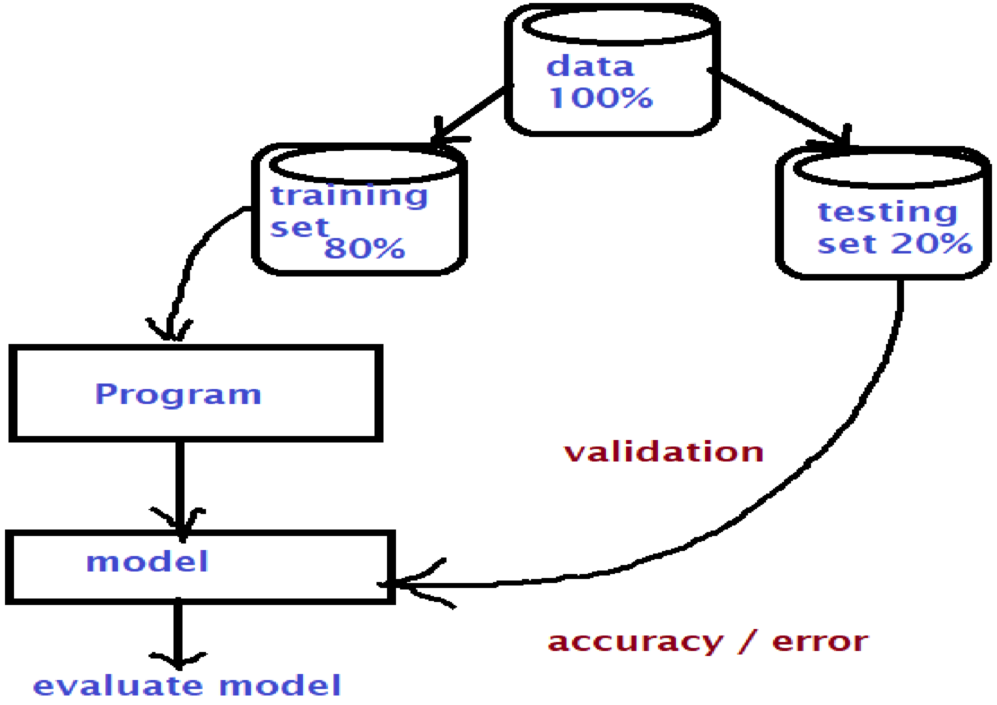


ACTUAL

		Negative	Positive
PREDICTION	Negative	TRUE NEGATIVE	FALSE NEGATIVE
	Positive	FALSE POSITIVE	TRUE POSITIVE

Get the data

- Decide the data source (file / database / online /api)
- Download the data and make it available for the further learning
- Take a Quick Look at the Data Structure
  - Understand the data set (numeric / textual / categorical etc) and understand its features /columns /variables
  - Evaluate the features and decide which one(s) are needed(dependent/independent→ correlation analysis)
- **Create a Test Set**
  - Keep some records aside for testing and validation



Discover and Visualize the Data to Gain Insights

- Visualize the data
  - Use libraries like matplotlib or seaborn
  - Understand the pattern and relationship
- Look for correlation
- Experiment with attribute combinations

Prepare the Data for Machine Learning Algorithms

- Data Cleaning
    - Process of cleaning the data set to prepare it for ML algorithm
    - Steps
        - Check for the missing data(NA values)
        - Check for wrong data types
        - Add features if needed
        - Remove unwanted features
- Feature Scaling
    - ML algorithms don’t perform well when the input numerical attributes have very different scale
    - Scale the features to bring all of them to a single scale
- Handle categorical / text data
    - Use transformers to convert categorical to numerical (eg. Label encoding / ordinal encoding, one-hot encoding etc)

Select and Train a Model

- Training the model using train data set
    - Create a model using selected algorithm
    - Save the model for future use
- Evaluation the model
    - Evaluate the model to see if there is any chance to improve the accuracy
    - Techniques
        - Cross Validation

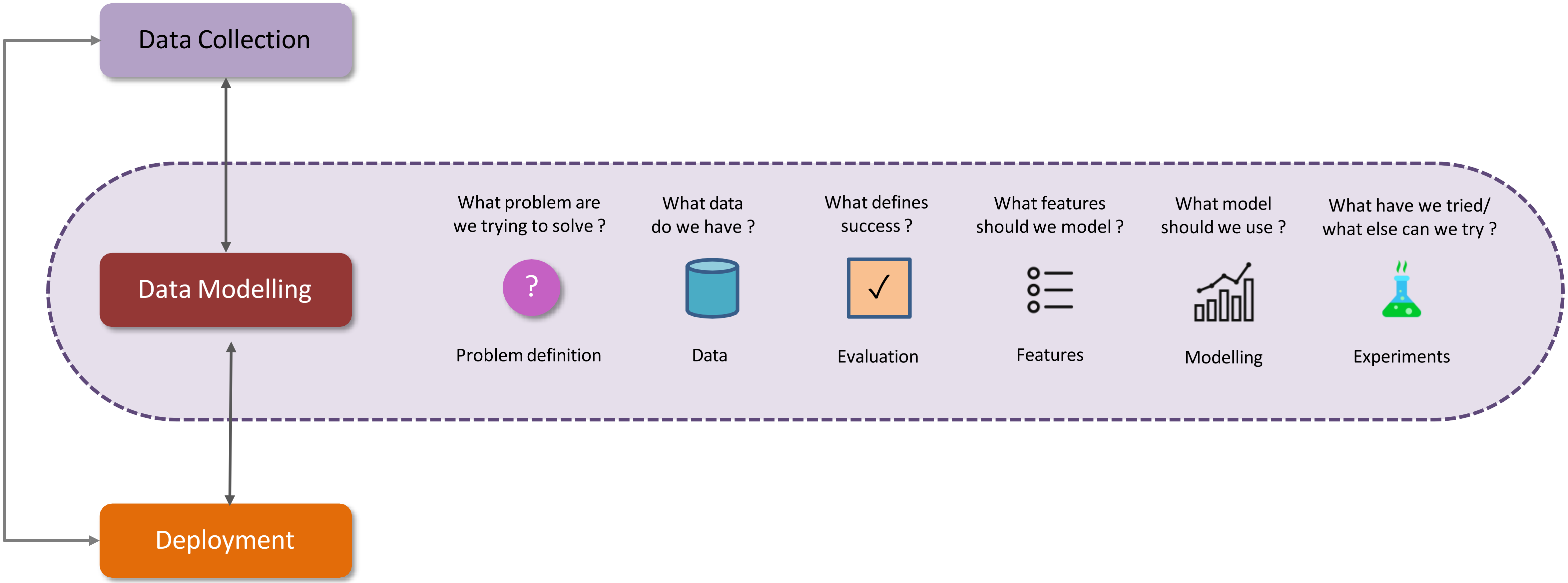
Fine-Tune Your Model

- **Grid Search**
  - One option would be to fiddle with the hyperparameters manually, until you find a great combination of hyper parameter values(configuration of algorithms)
  - This would be very tedious work, and you may not have time to explore many combinations
  - You can also automate this process using libraries like sci-kit
- **Randomized Search**
  - The grid search approach is fine when you are exploring relatively few combinations
  - But when the hyperparameter search space is large, it is often preferable to use randomized search
- **Ensemble Methods**
  - Another way to fine-tune your system is to try to combine the models that perform best
  - The group (or "ensemble") will often perform better than the best individual model, especially if the individual models make very different types of errors.
- Analyse the Best Models and Their Errors
- Evaluate Your System on the Test Set

Launch, Monitor, and Maintain Your System

- Deploy the application for the end users → production
- Monitor the application’s performance
- If the data keeps evolving, update your datasets and retrain your model regularly
- You should probably automate the whole process as much as possible
  - Collect fresh data regularly and label it
  - Write a script to train the model and fine-tune the hyperparameters automatically. This script could run automatically, for example every day or every week, depending on your needs
  - Write another script that will evaluate both the new model and the previous model on the updated test set, and deploy the model to production if the performance has not decreased (if it did, make sure you investigate why)

Summary



Data Collection

Data Modelling

Deployment

What problem are we trying to solve ?

Problem definition

What data do we have ?

Data

What defines success ?

Evaluation

What features should we model ?

Features

What model should we use ?

Modelling

What have we tried/ what else can we try ?

Experiments

Discrete Variable

- A discrete variable is a type of variable that can only take on specific, distinct values.
- These values are typically whole numbers or integers.
- Discrete variables often represent counts or categories.
- eg: Number of students in a classroom:
- It is a discrete variable because it can only take on whole
- number values (e.g., 25 students, 30 students).

- Outcomes of rolling a six-sided die:
- The output will be (1, 2, 3, 4, 5, or 6) which are discrete because
- they consist of distinct, separate categories.

- Number of books on a shelf:
- The number of books is discrete because it cannot take on
- fractional or continuous values
- (e.g., 5 books, 10 books, 15 books).

Continuous Variable

- Continuous variable is a type of variable that can take on any value within a given range.
- continuous variables can represent an infinite number of possible values, including fractional and decimal values.
- Continuous variables often represent measurements or quantities.
- Eg:
- Height: Height is a continuous variable because it can take on any value within a range (e.g., 150.5 cm, 162.3 cm, 175.9 cm).

- Weight: Weight is continuous because it can be measured with precision and can take on any value within a range
- (e.g., 55.3 kg, 68.7 kg, 72.1 kg).

- Time: Time can be measured with precision, and it can take on any value (e.g., 10:30:15.5 AM, 10:45:30.75 AM).

Categorical Data

- Categorical data refers to variables that belong to distinct categories such as labels, names or types.

- Since most machine learning algorithms require numerical inputs, encoding categorical data to numerical data becomes important.

- Proper encoding ensures that models can interpret categorical variables effectively, leading to improved predictive accuracy and reduced bias.
- .

Types of Categorical Data

1. Nominal Data: Nominal data consists of categories without any inherent order or ranking. These are simple labels used to classify data.
Eg: 'Red', 'Blue', 'Green' (Car Color).
Encoding Options: One-Hot Encoding or Label Encoding, depending on the model's needs.

2. Ordinal Data: Ordinal data includes categories with a defined order or ranking, where the relationship between values is important.
Eg: 'Low', 'Medium', 'High' (Car Engine Power).
Encoding Options: Ordinal Encoding.

Using the right encoding techniques, we can effectively transform categorical data for machine learning models which improves their performance and predictive capabilities

Thank You!!