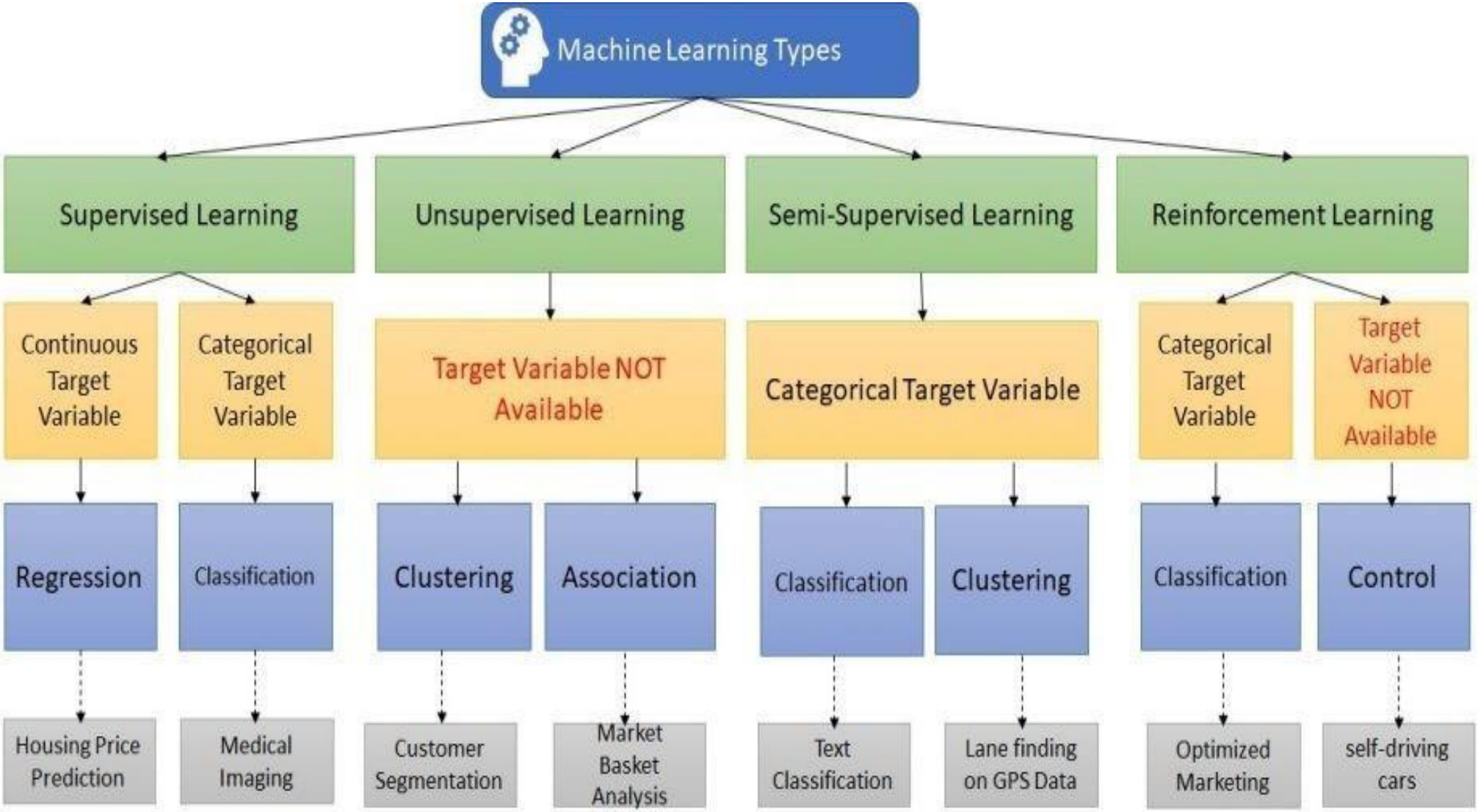
Classification vs Regression

(Linear Regression Vs Logistic Regression)

Dr Gopal Jamnal

Types of Machine Learning Algorithms

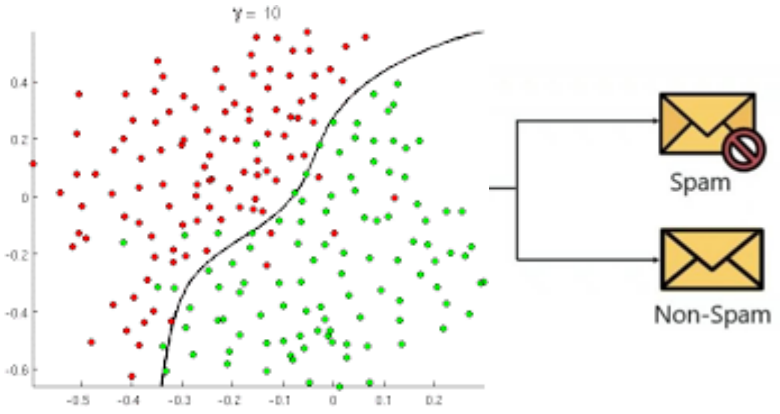


Classification vs Regression

Classification

Classification is the task of predicting the categorical label

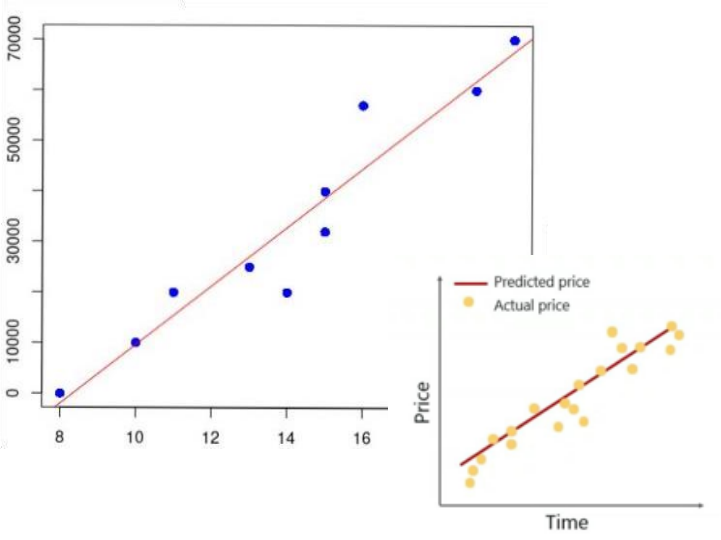
- In classification problem data is classifieds into two or more **classes**
- A classification problem with **two** classes is called **binary classification**, problem with more than two classes is called **multi-class classification**



Regression

Regression is the task of predicting a continuous quantity

- A regression problem requires a **prediction of a quantity**
- A regression problem with multiple input variable is called multivariate regression analysis



Multiple Linear Regression

Multiple linear regression (MLR) is a statistical technique that uses several independent variables to predict the outcome of a response (dependent) variable.

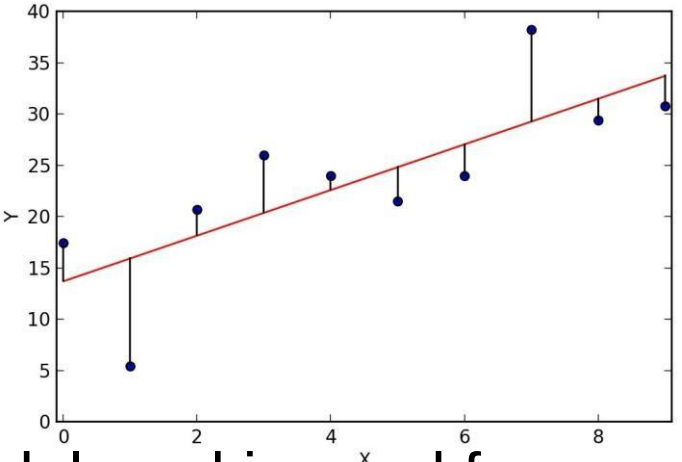
$$y=\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots + \beta_n x_n + e$$

- y - dependent variable
- X - explanatory variables
- β - intercept (constant term)
- β_i - slope coefficients for each explanatory variable
- e - the model's error (residual)

Root Mean Squared Error

Root mean squared error, or **RMSE** is defined as follows:

$$RSME = \sqrt{\frac{\sum (y - \hat{y})^2}{n}}$$



It measures the **overall accuracy** of the model, and is used for comparing it models (including those fitted using machine learning techniques)

Residual Standard Error

Residual standard error is defined:

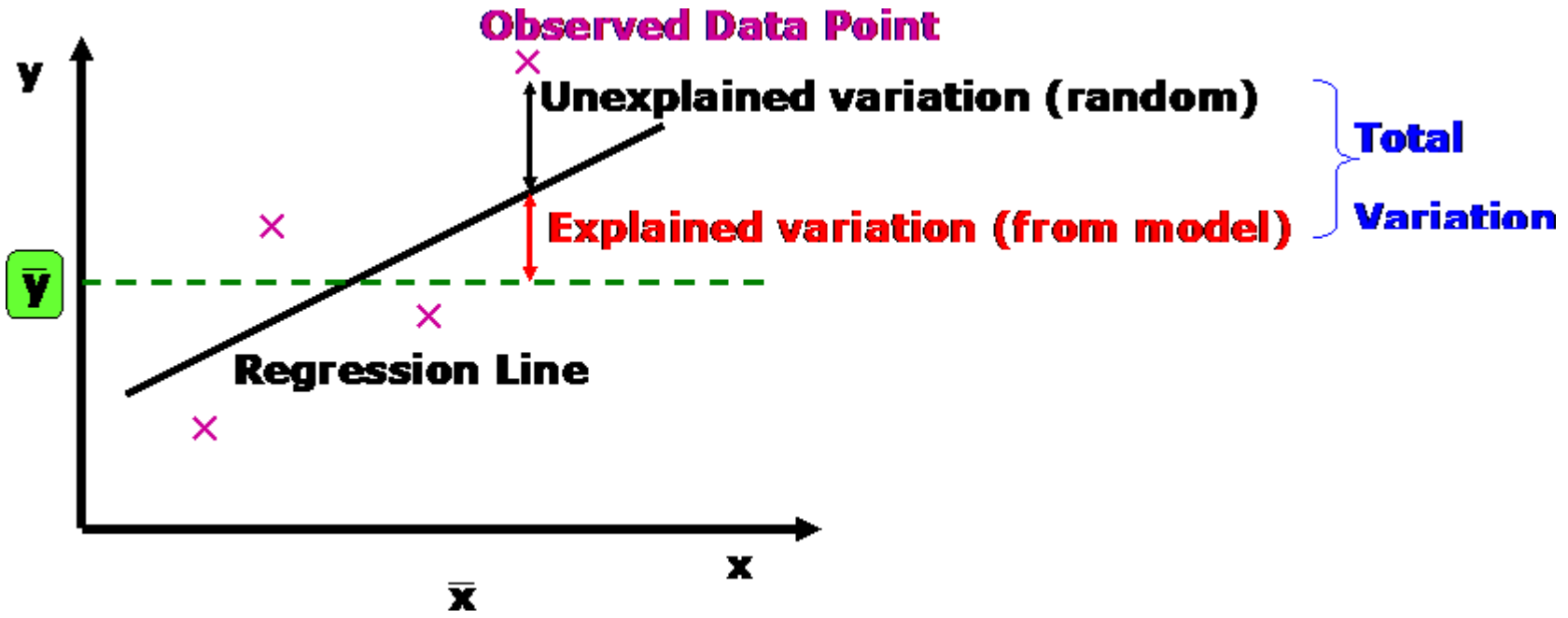
$$RSE = \sqrt{\frac{\sum(y - \hat{y})^2}{n - p - 1}}$$

- represents **average variation** of the observations points around the fitted regression line
- it is the **standard deviation** of residual errors
- a good measure for **comparing models,** the smaller error the better
- defines an average number of units for which the observed value y deviate from the true regression line
- RSE/mean - the **percentage error**

R^2 – Coefficient of Determination

R^2 - coefficient of determination, also called the **R-squared** statistic is defined:

$$R^2 = 1 - \frac{\sum(ExplainedVariation)^2}{\sum(VariationfromMean)^2} = 1 - \frac{\sum(\hat{y} - \bar{y})^2}{\sum(y_i - \bar{y})^2}$$



R^2 – Coefficient of Determination

R^2 - coefficient of determination, also called the **R-squared** statistic is defined:

$$R^2 = 1 - \sqrt{\frac{\sum(y - \hat{y})^2}{\sum(y - \bar{y})^2}}$$

- it ranges from 0 to 1
- represents the **proportion of information** (i.e. variation) in the data that can be **explained by the model**
- measures, how well the model fits the data
- a value close to 1 indicates that a large proportion of the variability in the outcome has been explained by the regression model
- a value near 0 indicates that the regression model did not explain much of the variability in the outcome

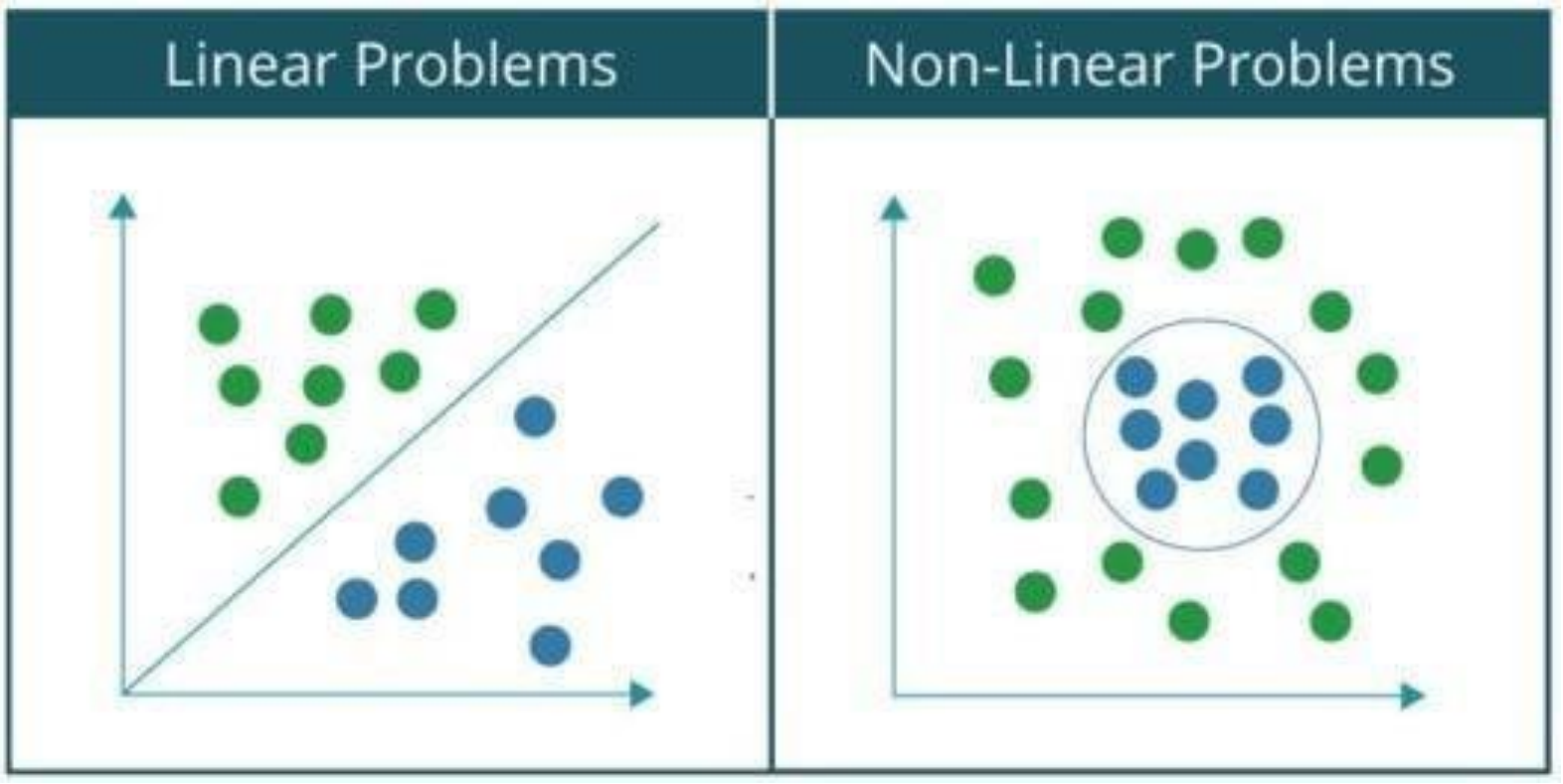
Interpreting R^2

- Before you look at the statistical measures for goodness-of-fit, you should check the **residual plots**. Residual plots can reveal unwanted residual patterns that indicate biased results more effectively than numbers
- R-squared **cannot** determine whether the coefficient estimates and predictions are biased, which is why you must assess the residual plots.
- R-squared **does not indicate** whether a regression model is adequate. You can have a low R-squared value for a good model, or a high R-squared value for a model that does not fit the data!

Classification Problems

- Is the email an attempt at phishing?
- Is the customer likely to churn?
- Is the web user likely to click on advertisement?
- Is it a fraud?
- Is the patient likely to have a cancer?
- Is it a gun?
- Is the chemical compound toxic?
- Is this part likely to be faulty?

Classifiers Classification

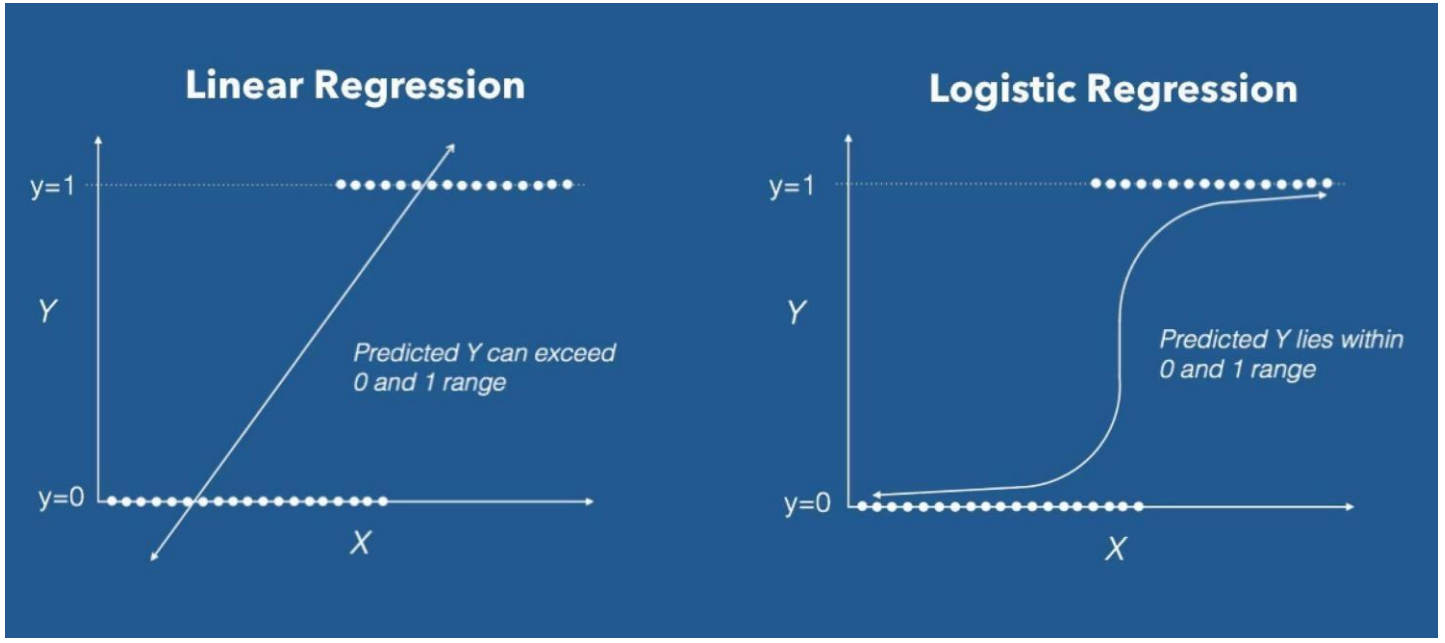


Linear Classifiers

- Logistic regression – probabilistic approach
- Fisher's linear discriminant – based on discriminant functions
- Perceptron – based on discriminant functions.

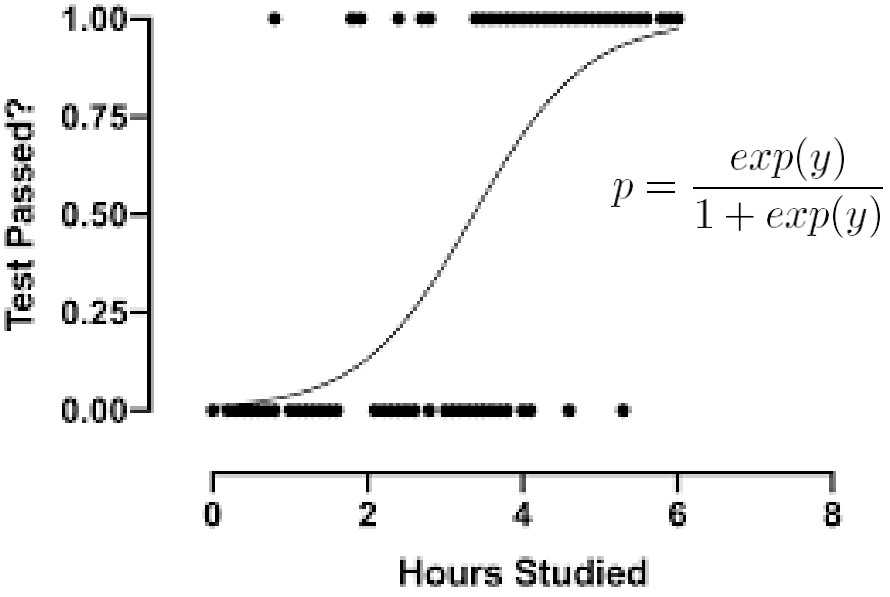
Logistic Regression

- The logistic model (or logit model) is used to model the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or health/sick.
- It predicts **TRUE or FALSE** instead of something continuous
- It fits **S-shape** “logistic function”
- The curve tells **the probability of an event** based on some variables



Logistic Regression

- **Dependent variable Y** - the response binary variable: eg. Test Passed: 0,1
- **Independent variable X** – the predictor variable used to predict response variable Y: e.g. hours studied



The following equation is used to represent logistic regression

$$\log[\frac{p}{1-p}] = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + ... + \beta_n * x_n$$

logarithm of the odd, also known as log-odd or logit.

Logistic Regression Interpretation

Lets consider a model that predicts obesity based on glucose level:

- Logistic formula:

```
model <- glm( diabetes ~ glucose, data = train.data, family = binomial)
summary(model)$coef
```

$$p = \frac{exp(-6.32 + 0.043 * glucose)}{1 + exp(-6.32 + 0.043 * glucose)}$$

```
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -6.3267     0.7241   -8.74 2.39e-18
## glucose       0.0437     0.0054    8.09 6.01e-16
```

- positive b coefficient indicates that increasing x will be associated with increasing p
- negative b coefficient indicates that increasing x will be associated with decreasing p
- one unit increase in the glucose concentration will increase the odds of being diabetes-positive by exp(0.043) 1.05 times.
- Z-test: Wald test to test he null hypothesis of the corresponding coefficient being zero.

Evaluating Classification Models

- **How good is your model?** - test the model performance on test/validation data or cross-validation (CV)
- Evaluation Metrics
  - Confusion Matrix
  - Classification accuracy
  - Precision, Recall, Specificity, F1-score
  - ROC curve, Area Under Curve (AUC)
  - PR curves

Confusion Matrix

The confusion matrix is a table showing the **number of correct and incorrect predictions** categorized by type of response.

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP **Type I Error**
	Negative	FN **Type II Error**	TN

Confusion Matrix

- **True Positive (TP)** is an outcome where the model correctly predicts the positive class.
- **False Negative (FN)** refers to the number of predictions where the classifier incorrectly predicts the positive class as negative.
- **False Positive (FP)** refers to the number of predictions where the classifier incorrectly predicts the negative class as positive.
- **True Negative (TN**) is an outcome where the model correctly predicts the negative class

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP **Type I Error**
	Negative	FN **Type II Error**	TN

Classification Metrics

		Actual: Positive	Actual: Negative	
Predicted	Positive	TP	FP **Type I Error**	Precision
	Negative	FN **Type II Error**	TN	Negative predictive value
		Sensitivity/Recall	Specificity	

- Sensitivity/Recall/True Positive Rate - TP/(TP+FN)
- Specificity/True Negative Rate - TN/(FP+TN)
- Precision - TP/(TP+FP)
- NPV (negative predictive value) - TN/(FN+TN)
- **Accuracy** **- (TP+TN)/(TP+FP+FN+TN)**
- Error - (FN+FP)(TP+FP+FN+TN)
- F-1 - 2*(Precision *Recall)/(Precision+ Recall)

Testing Accuracy

Let assume you have developed a binary classifier with accuracy > 80%

Do you think it is a good model?

Testing Accuracy

Let assume you have developed a binary classifier with accuracy > 80%

Assume sample distributions for classes:

- Class negative: 80%
- Class positive: 20%

Do you think you have a good model?

Testing Accuracy

Let assume you have developed a binary classifier with accuracy > 80%

Assume sample distributions for classes:

- Class negative: 80%
- Class positive: 20%

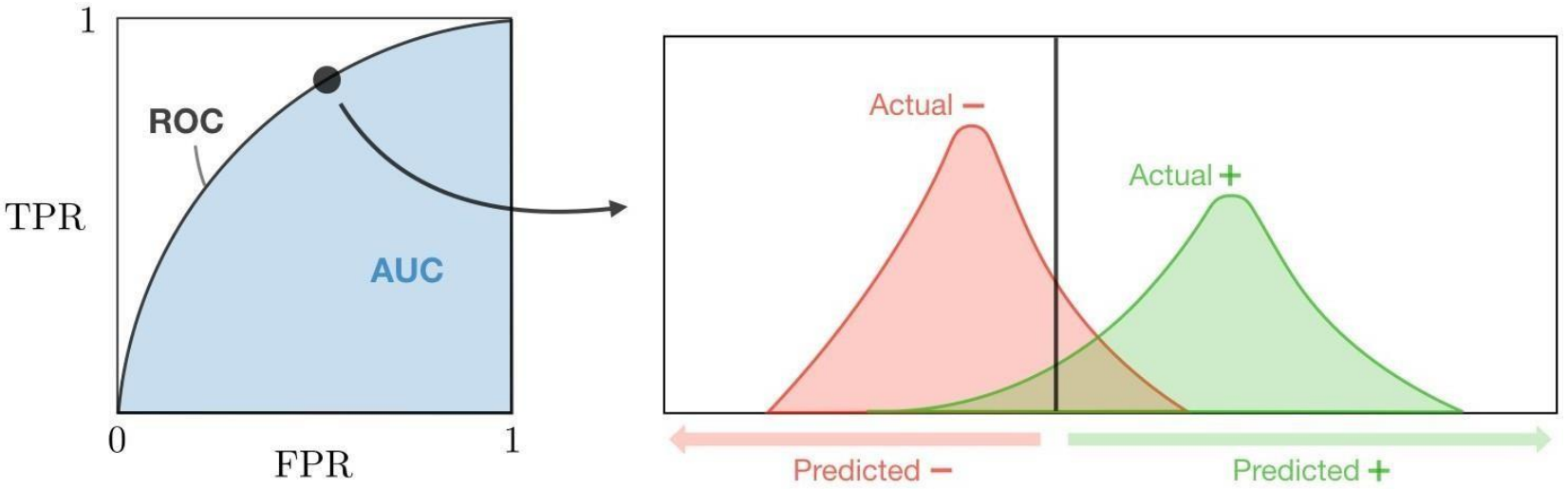
Assume your sensitivity is 0.

Do you think you have a good model?

ROC Curve

(**Receiver Operating Characteristics curve**) is a popular graphical measure for assessing the performance or the accuracy of a classifier, which corresponds to the total proportion of correctly classified observations.

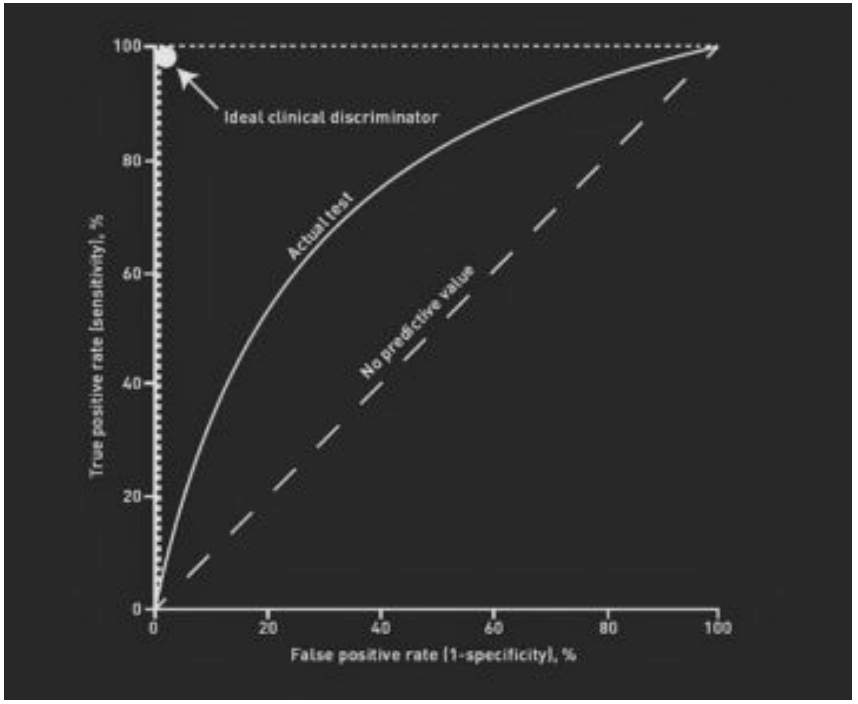
Classifier has to predict score that reflects the degree to which an instance belongs to one class rather than another class.



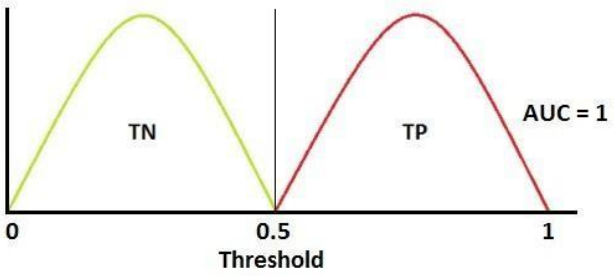
False Positive Rate (FPR)= 1-Specificity
True Positive Rate (TPR) = Sensitivity

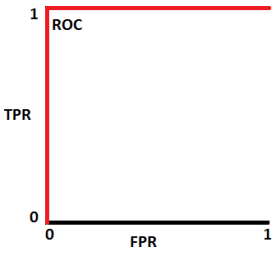
Interpreting ROC

- ROC curve shows the trade-off between sensitivity (or TPR) and specificity.
- Classifiers that give curves closer to the top-left corner indicate a better performance.
- As a baseline, a random classifier is expected to give points lying along the diagonal (FPR = TPR).
- The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test.

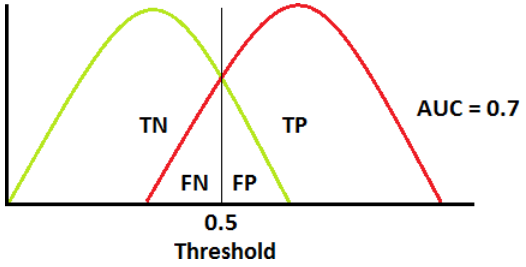


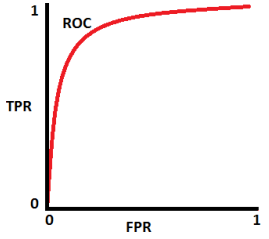
Interpreting ROC



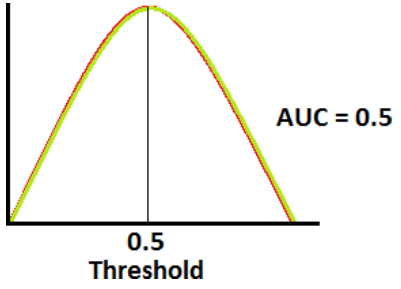


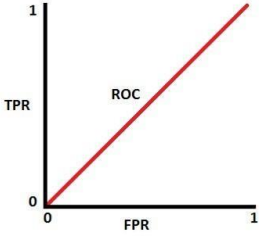
- Correctly identify positive and negative classes



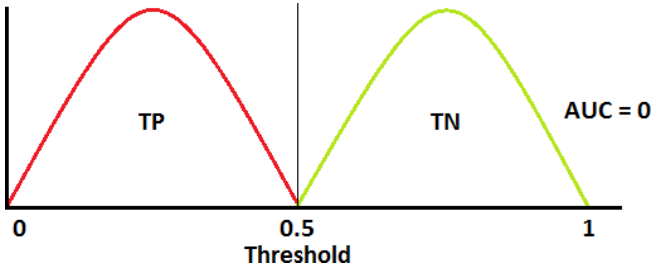


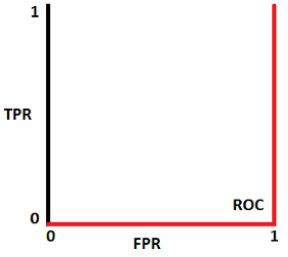
- Type I and Type II errors present





- The worst situation. Model has no discrimination capacity to distinguish between positive class and negative class.





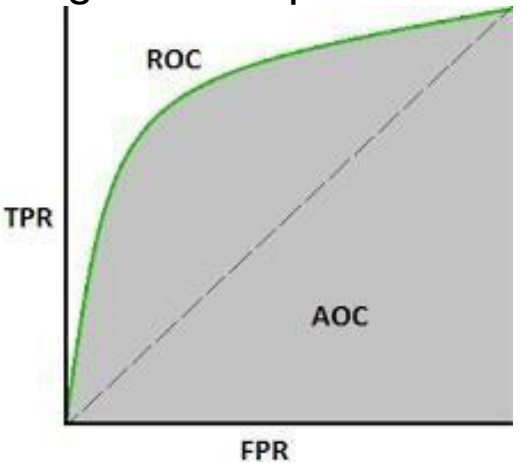
- Model is actually reciprocating the classes. It means the model is predicting a negative class as a positive class and vice versa.

Area Under Curve

The Area Under the Curve (AUC):

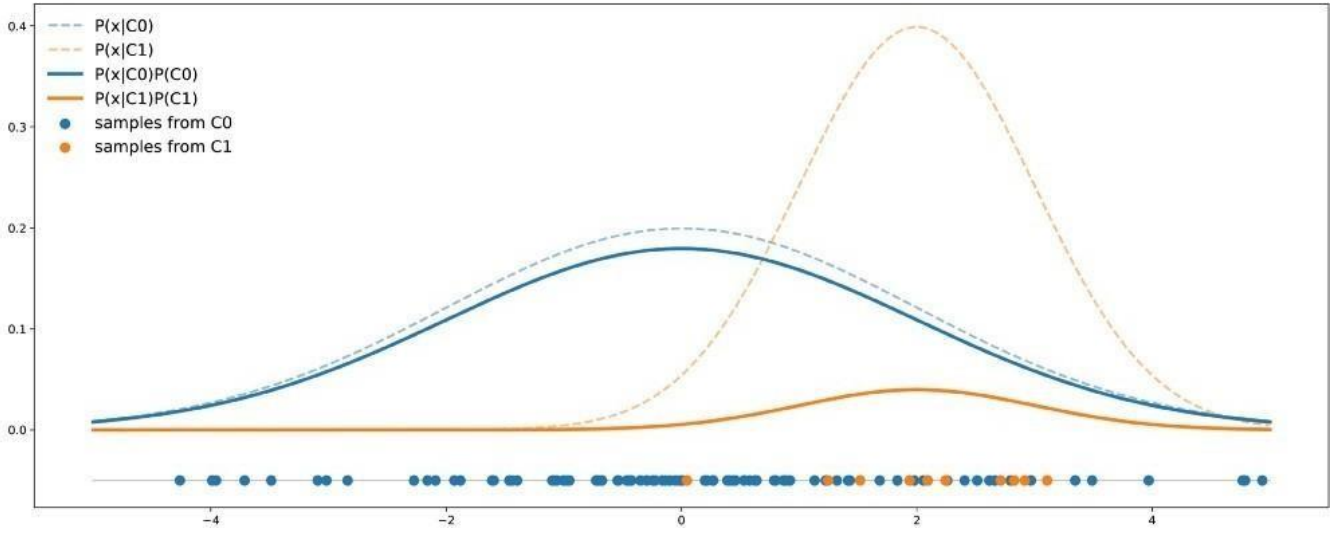
- **summarizes the overall performance** of the classifier, over all possible probability cut-offs
- **represents the ability** of a classification algorithm to **distinguish** positives from negative classes
- varies between **0.50 (random classifier) and 1**
- value **above 0.80** is an indication of a **good classifier**
- can be used to **compare different classifiers**. Classifiers with higher AUC perform better

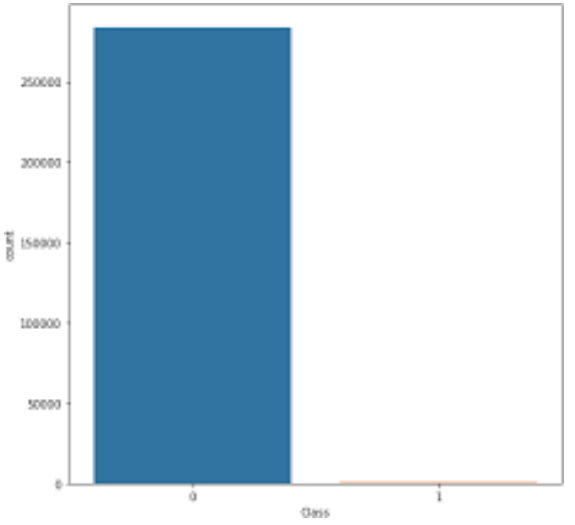
$$AUC = \int_0^1 TPR(x)dx, \text{ where x} \in FPR$$



Imbalance Data

- Data where sample **distribution across the known classes is biased or skewed**
- It is a challenge for predictive modelling as most of methods used for classification were designed around the assumption of an equal number of examples for each class



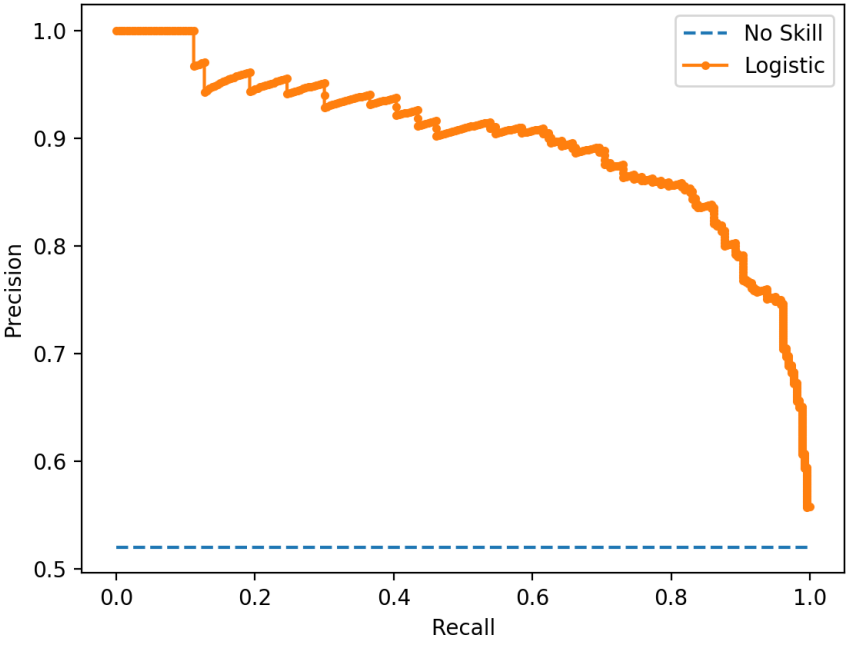


Strategies for Imbalance Data

- **Other performance Metrics –** precision, recall, F1 score
- **Change algorithm** – find one that perform well on imbalance data
- **Undersampling (downsampling)** - consists in sampling from the majority class in order to keep only a part of these points
- **Oversampling (upsampling)** - consists in replicating some points from the minority class in order to increase its cardinality
- **Generating synthetic data -** consists in creating new synthetic points from the minority class (**SMOTE** method) to increase its cardinality

Precision-Recall Curve

- ROC curves can be misleading in some very imbalanced applications
- A ROC curve can still look good (ie better than random) while misclassifying most or all of the minority class
- Precision-Recall (PR) curves are specifically tailored for the detection of rare classes and are useful in those scenarios
- A PR curve will show that your classifier has a low performance if it is misclassifying most or all of the minority class



Multi-Class Evaluation

- It is extension of the binary classifiers
  - a collection of true vs predicted binary class, one per class
- Overall evaluation matrices are average across classes
  - different ways to average the results

Multi-Class Evaluation

- It is extension of the binary classifiers
  - a collection of true vs predicted binary class, one per class
  - Confusion matrices are helpful here
- Overall evaluation matrices are average arccos classes
  - different ways to average the results
  - the support (number of instance) for each class is important to consider
- Multi-class label each instance has a multiple labels (not covered here)

Multi-Class Confusion Matrix

Extension of Binary Matrix

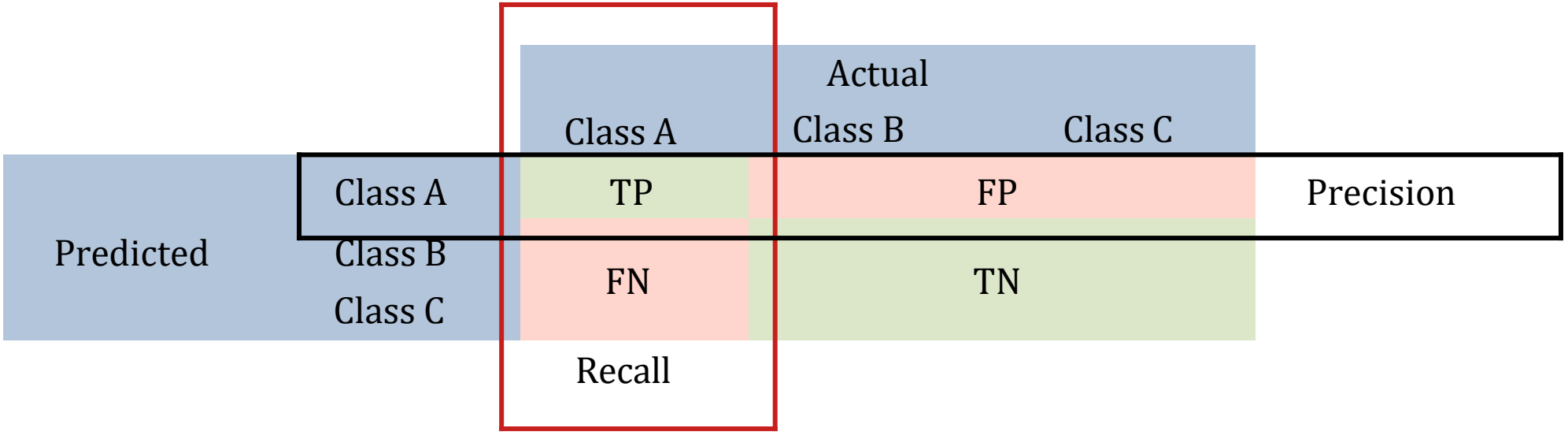
		Actual		
		Class A	Class B	Class C
Predicted	Class A	20	3	0
	Class B	0	10	1
	Class C	1	2	12

Accuracy = all correctly classified / all instances

Error = all wrongly classified / all instances

Multi-Class Confusion Matrix

Extension of Binary Matrix One vs All



- Precision_A = TP/(TP+FP)
- Recall_A = TP/(TP+FN)
- F-1_A = 2*(Precision_A *Recall_A)/(Precision_A + Recall_A)

It is repeated for all classes!

Macro vs Micro Average

Macro Average

- Compute metrics within each class
- Average resulting matrices across classes

$$PRE_{macro} = \frac{\sum_i^n PRE_i}{n}$$

Micro Average

- Aggregate outcomes across all classes
- Compute metrics with aggregate outcomes
- Each instance has equal length, larger classes have more influence

$$PRE_{micro} = \frac{\sum_i^n TP_i}{\sum_i^n TP_i + \sum_i^n FP_i}$$

Macro vs Micro Average

- If the classes have the same number of instances the micro- and macro-averages will be about the same
- If some classes are much larger than other you may want to:
  - weight you metrics towards the larger one – use micro-average
  - Weight your metrics toward a smaller one – use macro-average
- If the micro-average is much smaller than macro-average, examine larger classes for poor a metric performance
- If the macro–average is much smaller than micro-average, examine smaller classes for a poor metric performance