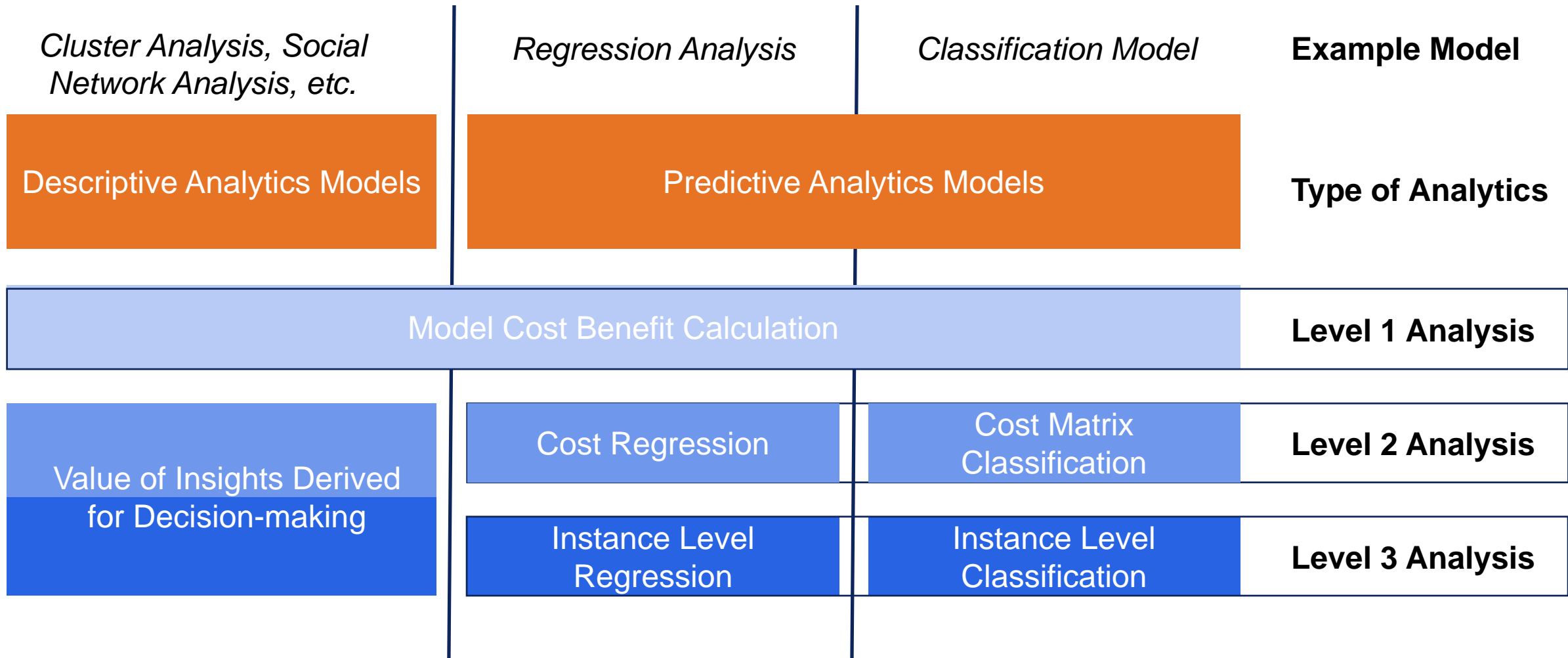


Model Cost Benefit Analysis Framework – Level 3



DARDEN SCHOOL of BUSINESS
McINTIRE SCHOOL of COMMERCE

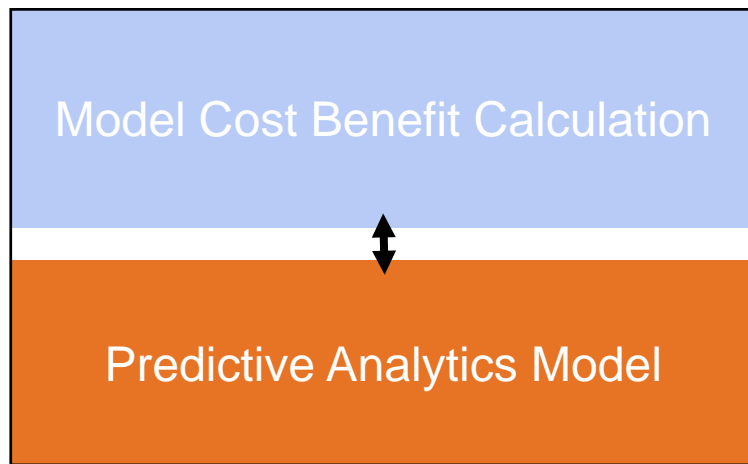
Model Cost-benefit Analysis Framework



Model Cost Benefit Analysis Framework – Predictive Analytics

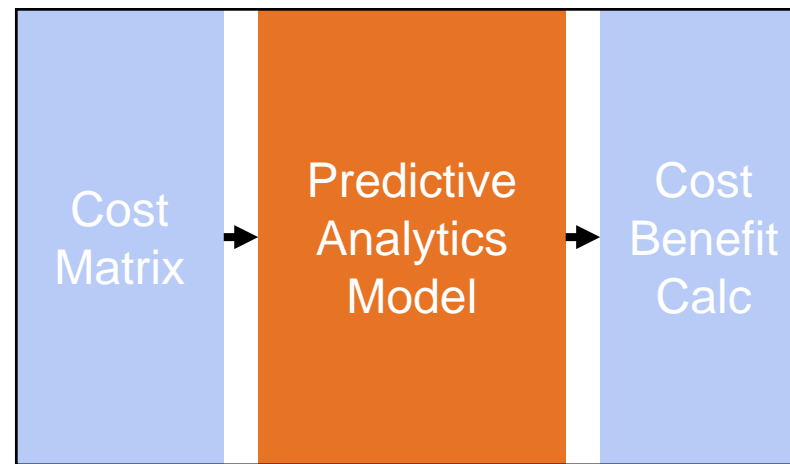
Three potential levels of analysis, depending on the problem context

Lv 1 – Model Level



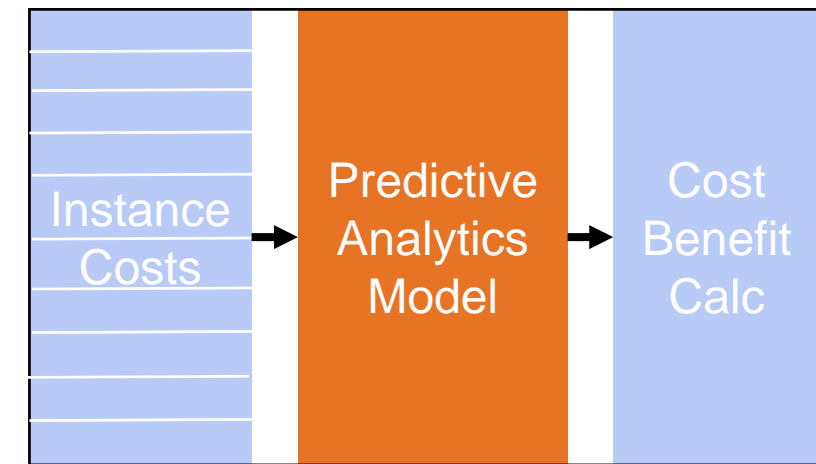
Treat model like a black box
Couple TP/TN/FP/FN with cost-benefit calculations

Lv 2 – Matrix Level



Input cost matrix ratios into model
Couple TP/TN/FP/FN with cost-benefit calculations

Lv 3 – Instance Level



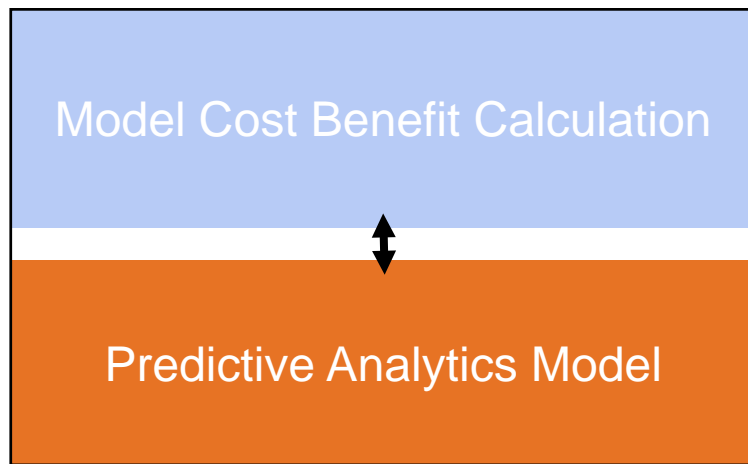
Input instance-level costs into model
Couple TP/TN/FP/FN with cost-benefit calculations



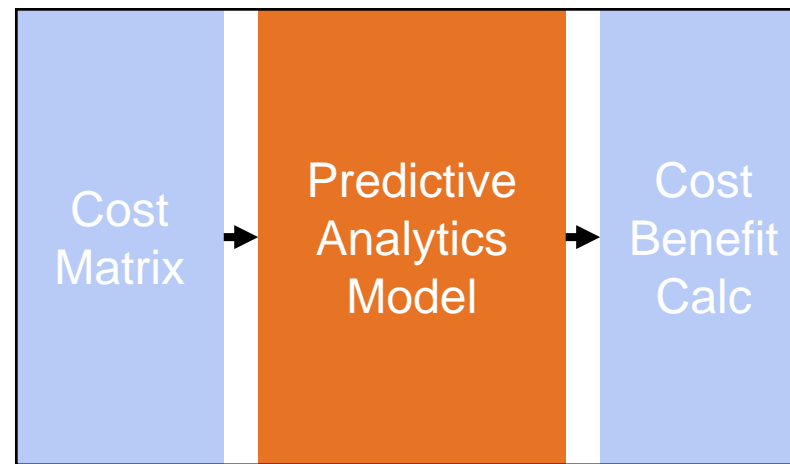
Model Cost Benefit Analysis Framework – Predictive Analytics

Three potential levels of analysis, depending on the problem context

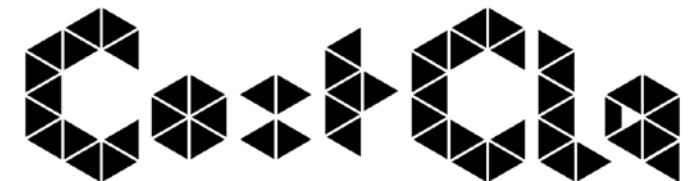
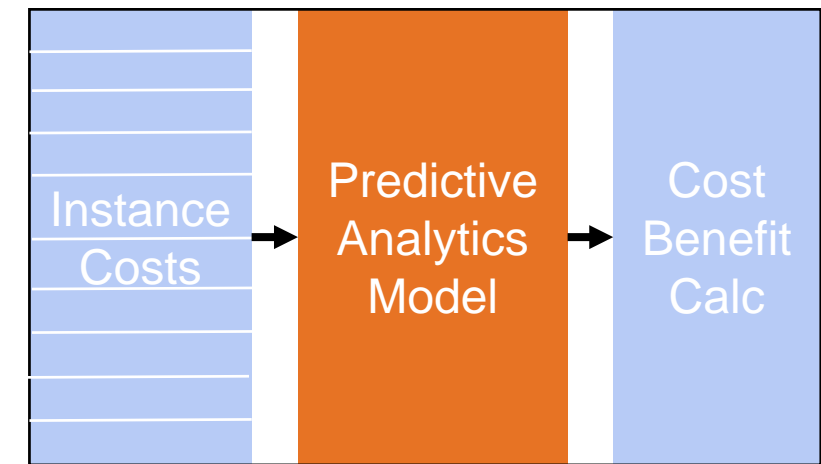
Lv 1 – Model Level



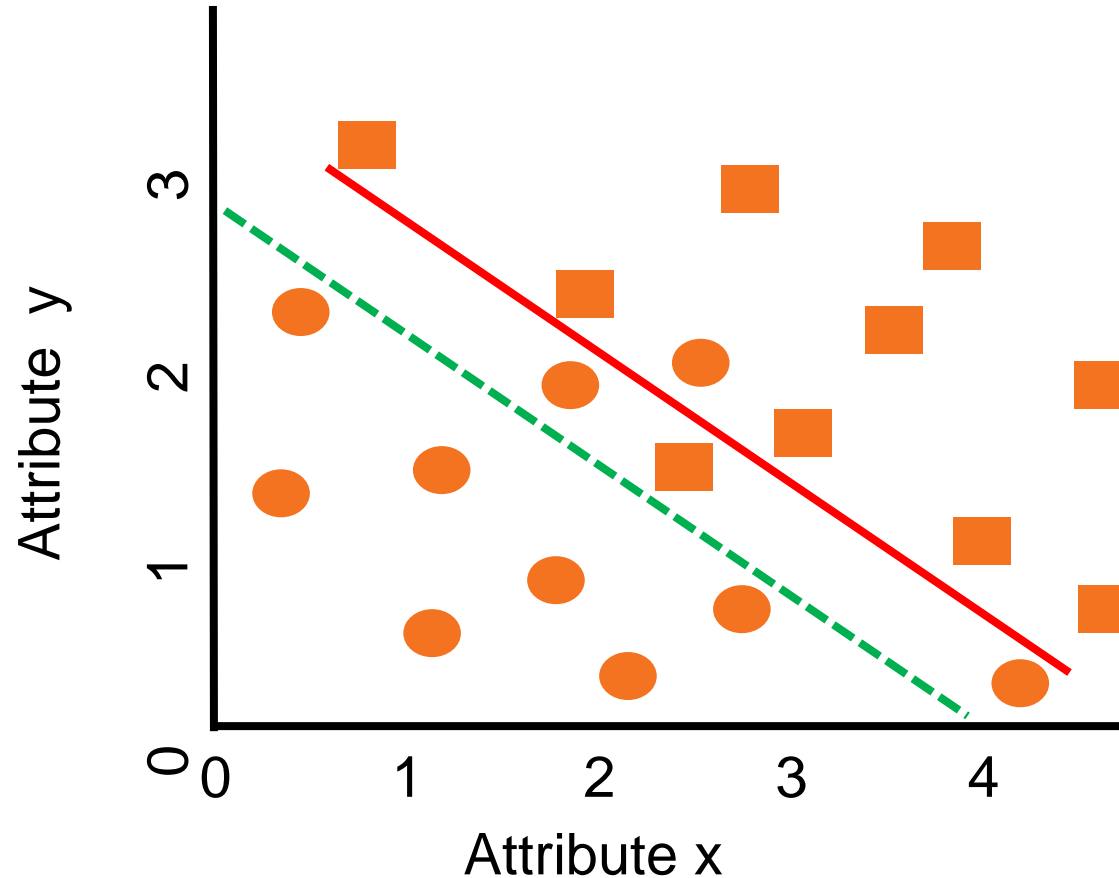
Lv 2 – Matrix Level



Lv 3 – Instance Level



Model Cost-benefit Analysis Framework – Asymmetric Costs – Lvl 2



Draw a single straight line that can best separate circles from squares (i.e., **minimal cost**)

Lv 2 Total Cost = 6

Lv 1 Total Cost = 7

Performance on Training Data:

Accuracy = $17/20 = 0.85$ **0.90**

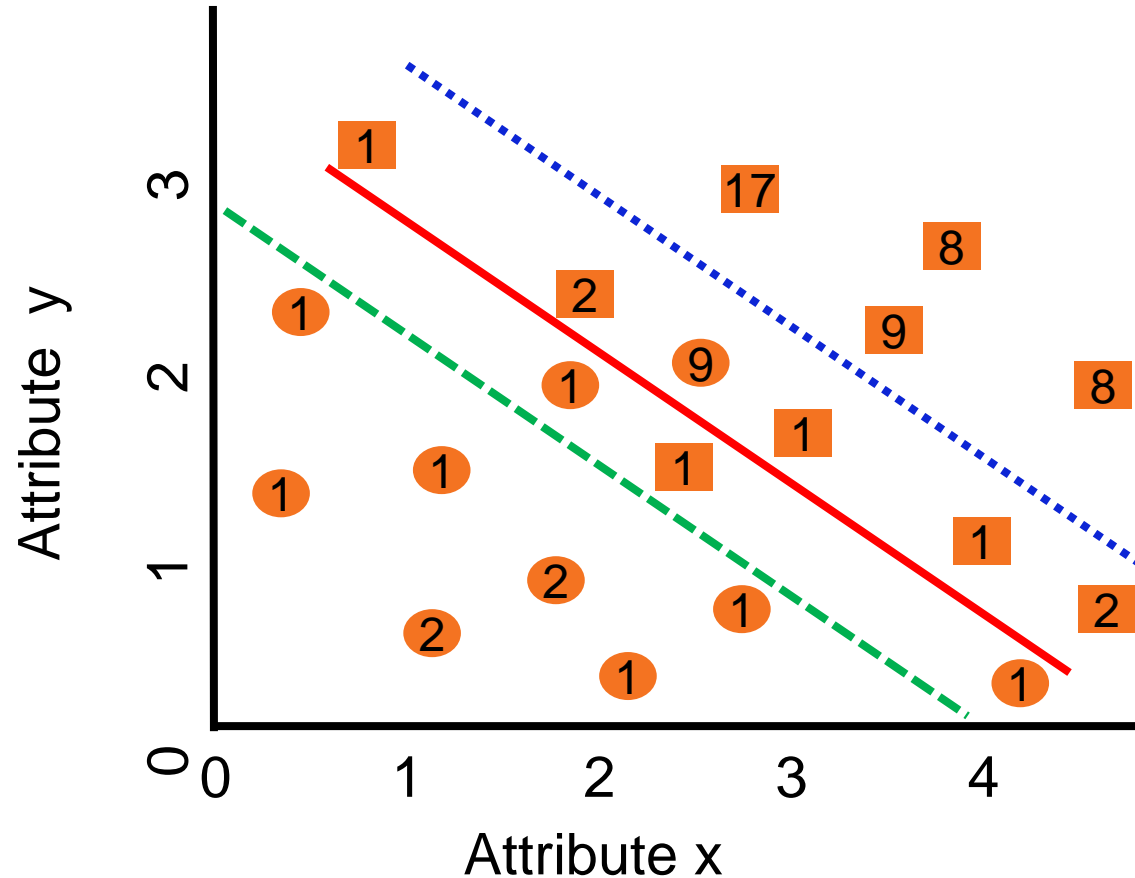
Circle Recall = $7/10 = 0.70$ **0.90**

Square Recall = $10/10 = 1.00$ **0.90**

Cost of False Sq = 2; Cost of False Cr = 5



Model Cost-benefit Analysis Framework – Asymmetric Costs – Lvl 3



Draw a single straight line that can best separate circles from squares (i.e., **minimal instance cost**)

Performance on Training Data:

Accuracy = $14/20 = 0.75$ **0.85** **0.90**

Circle Recall = $10/10 = 0.70$ **0.70** **0.90**

Square Recall = $4/10 = 0.40$ **1.00** **0.90**

Ave Cost of False Sq = 2; Ave Cost of False Cr = 5

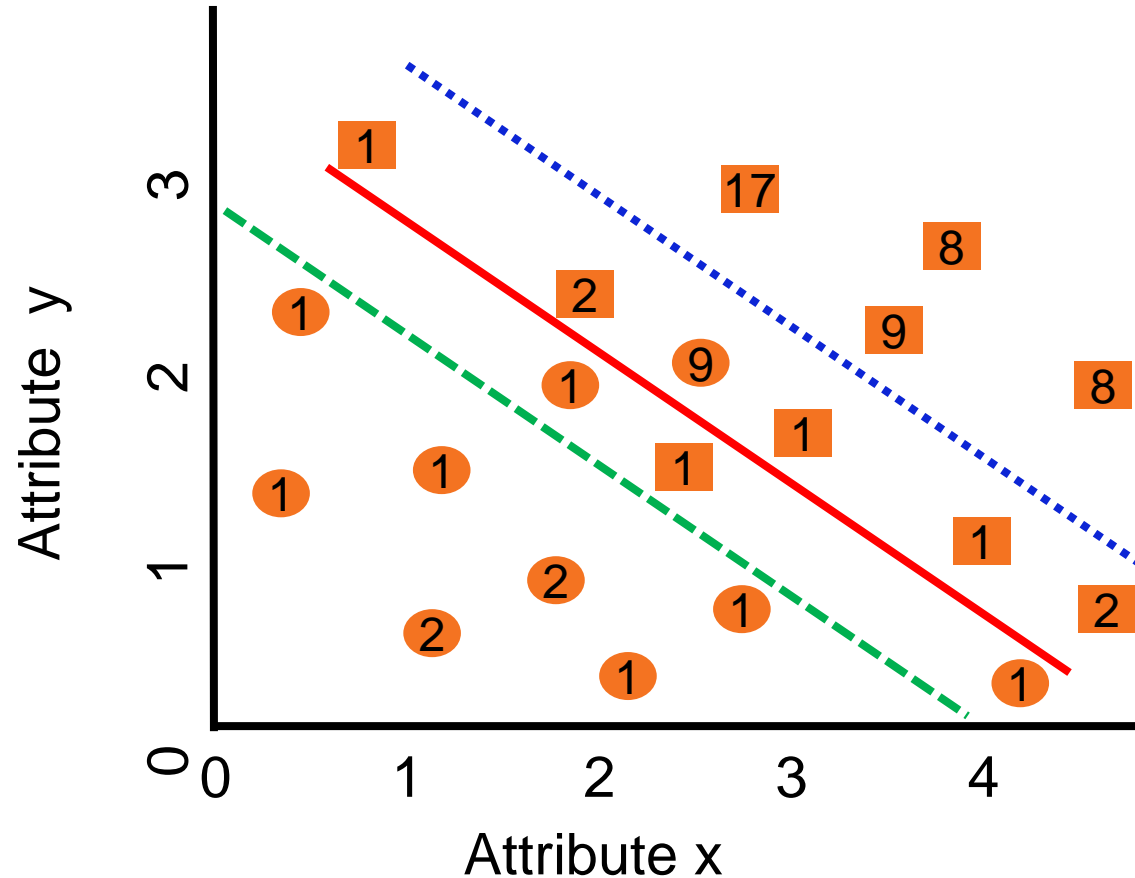
Lv 1 Total Cost = 10

Lv 2 Total Cost = 11

Lv 3 Total Cost = 8



Model Cost-benefit Analysis Framework – Asymmetric Costs – Lvl 3



Draw a single straight line that can best separate circles from squares (i.e., **minimal instance cost**)

Performance on Training Data:

Accuracy = $14/20 = 0.75$ **0.85** **0.90**

Circle Recall = $10/10 = 0.70$ **0.70** **0.90**

Square Recall = $4/10 = 0.40$ **1.00** **0.90**

Ave Cost of False Sq = 2; Ave Cost of False Cr = 5

Lv 1 Total Cost = 10

Lv 2 Total Cost = 11

Lv 3 Total Cost = 8

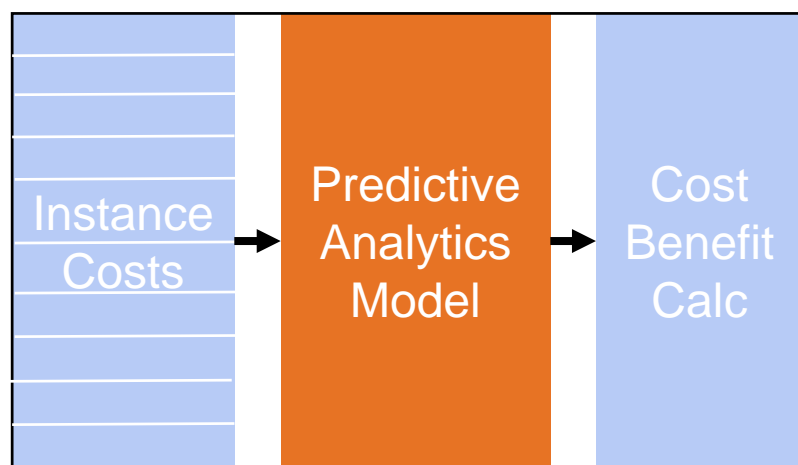


Model Cost-benefit Analysis Framework – Predictive Analytics Lv 3

Instance level analysis example #1 – **Cost sensitive Decision Tree**

Every instance has FP and FN costs

Lv 3 – Instance Level



```
++ COST MATRIX
Columns of cost matrix are: CostFalsePositive, CostFalseNegative, CostTruePositive
[ 1023.73054104 18750. 0. 0. ]
[ 694.27379722 5849.25 0. 0. ]
[ 719.67113704 6843.75 0. 0. ]
...,
[ 865.73457105 12563.25 0. 0. ]
[ 873.31929697 12860.25 0. 0. ]
[ 1013.6367265 18354.75 0. 0. ]]
```

```
def trainModelAndGetResults(classifier, trainData, testData, trainLabels, testLabels, costMatrixTrain,
                             costMatrixTest, classifierName):

    if classifierName=="CostSensitiveDecisionTree":

        # Train a cost-sensitive decision tree classifier
        classifier.fit(trainData, trainLabels, costMatrixTrain)

        # Get test scores
        predictions = classifier.predict(testData)
        predictionProbabilities = classifier.predict_proba(testData)
        predictionProbabilities = [item[1] for item in predictionProbabilities]
```

Use instance-level costs as input for training



