CART-ELC: Oblique Decision Tree Induction via Exhaustive Search

Andrew D. Laack

University of Wisconsin-Superior

May 13, 2025

Source Available:
`https://github.com/andrewlaack/cart-elc`

Problem

How can we determine if someone has diabetes based on their BMI?

Visualization

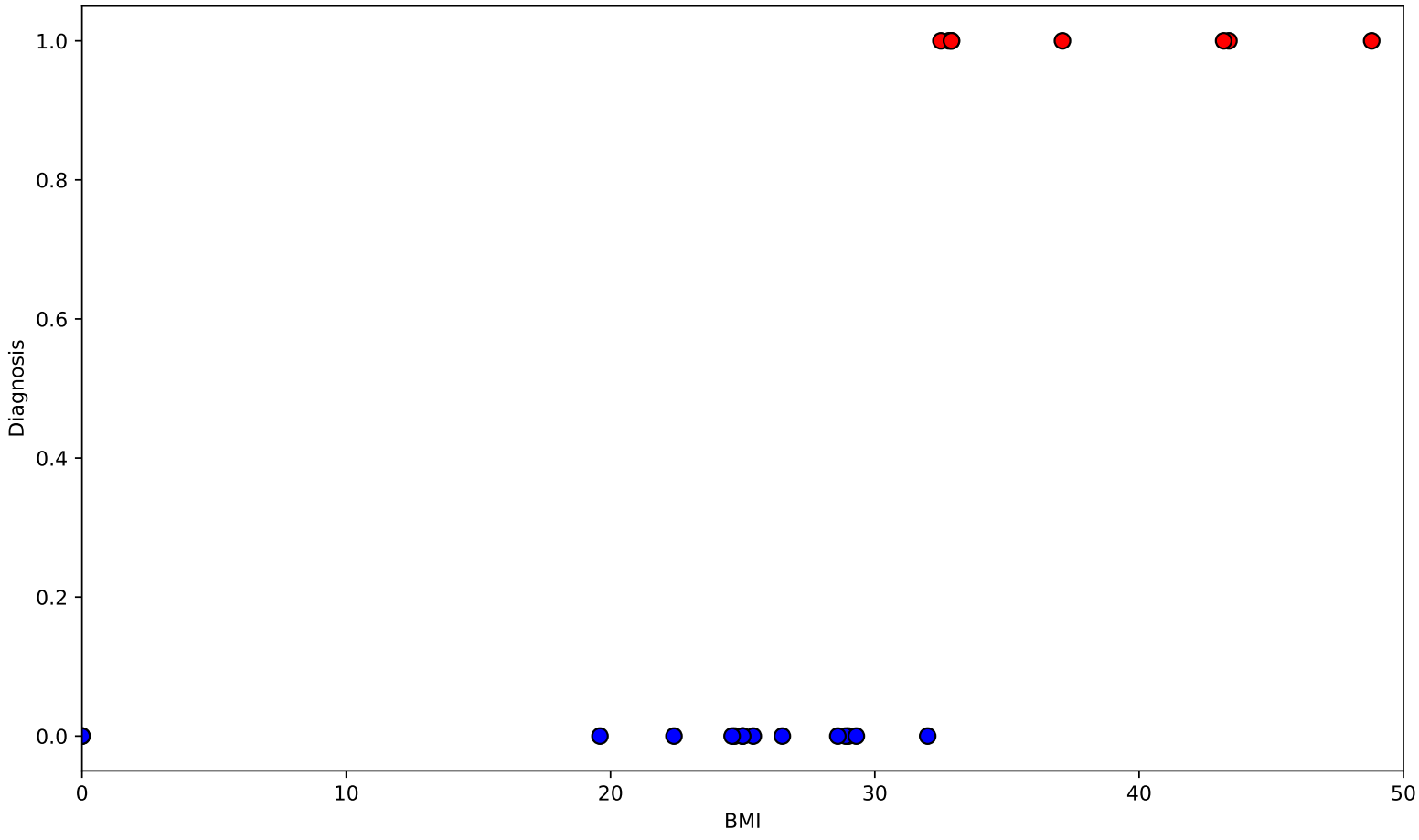

Figure: Diabetes Dataset [6] Graph

Solution

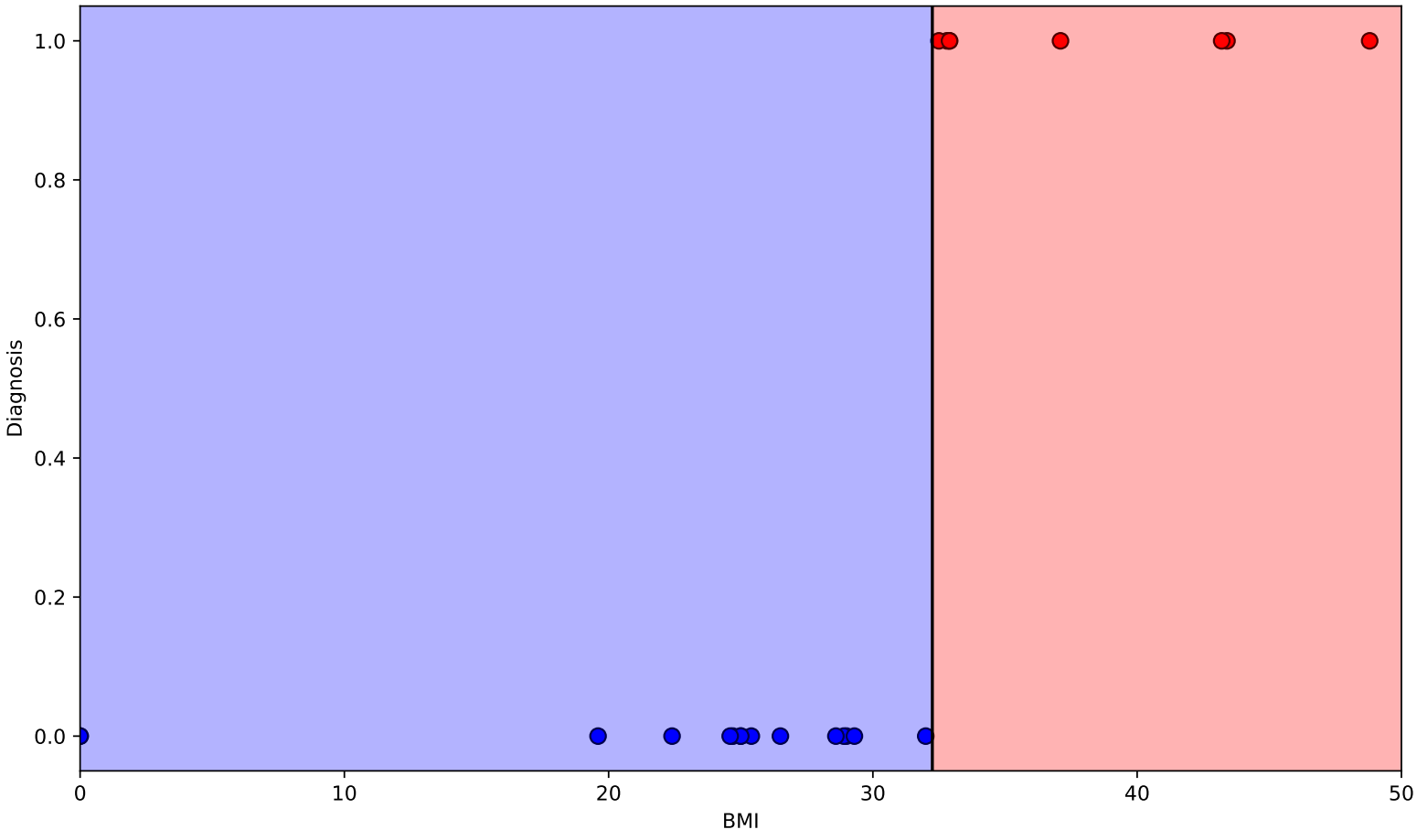

Figure: Diabetes Dataset Graph w/ Boundary

Tree

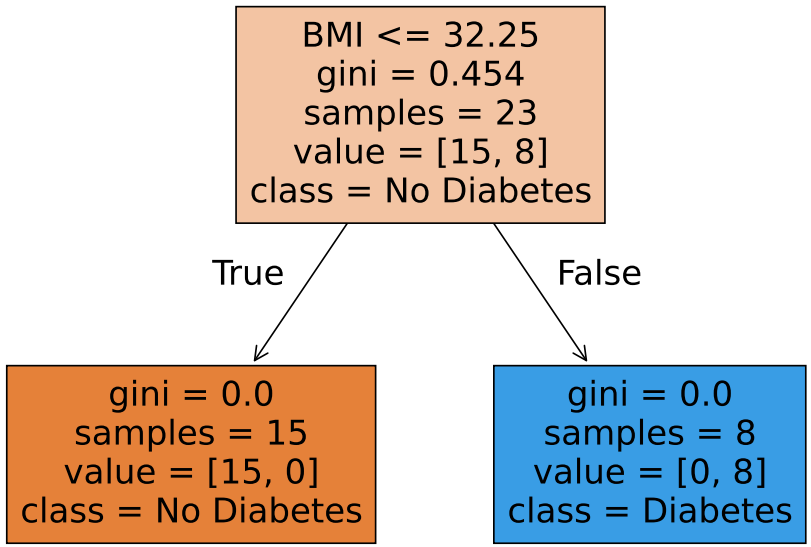

Figure: Tree Representation

But...

62.2% Accuracy

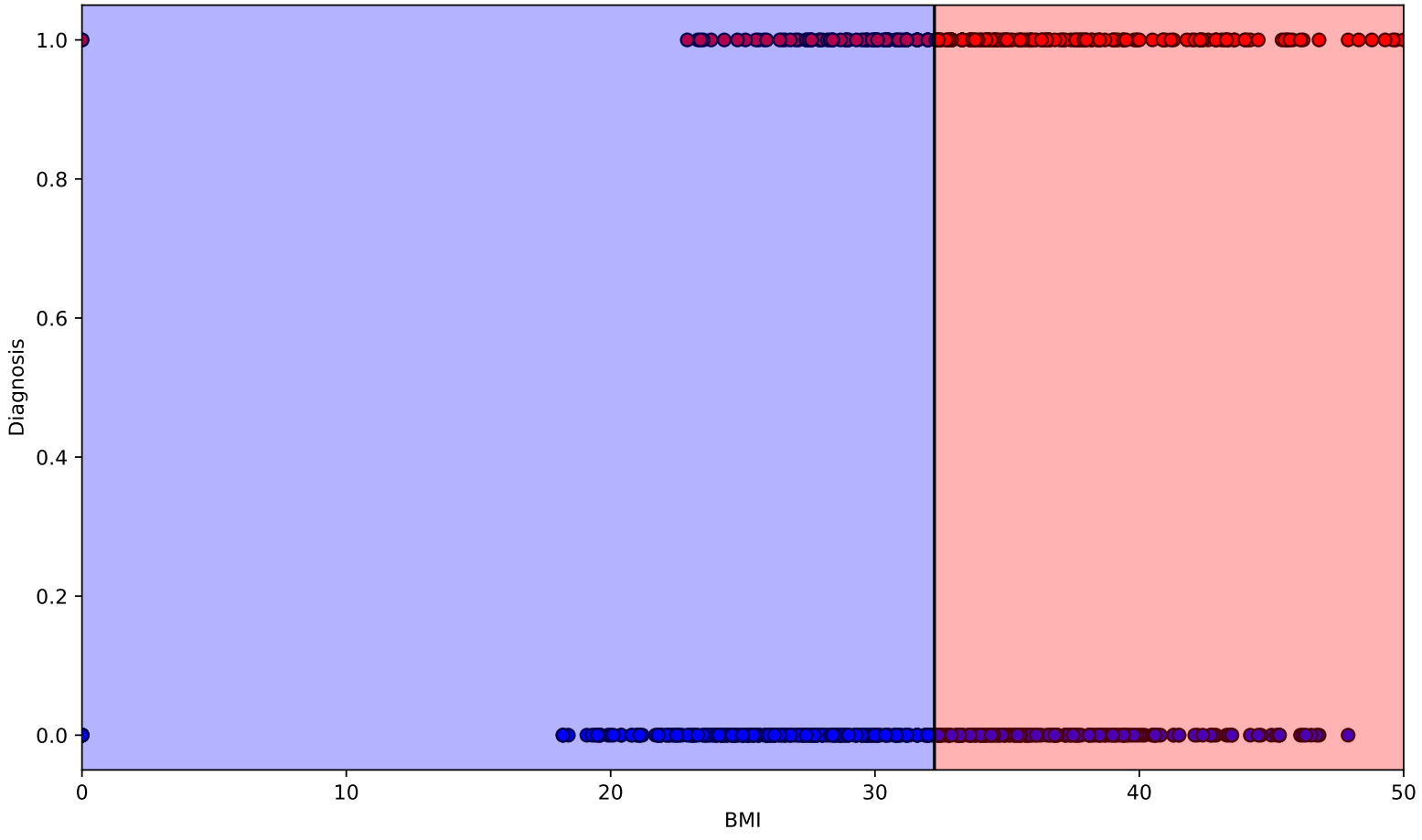

Figure: Full Diabetes Dataset Graph w/ Boundary

Extra Feature

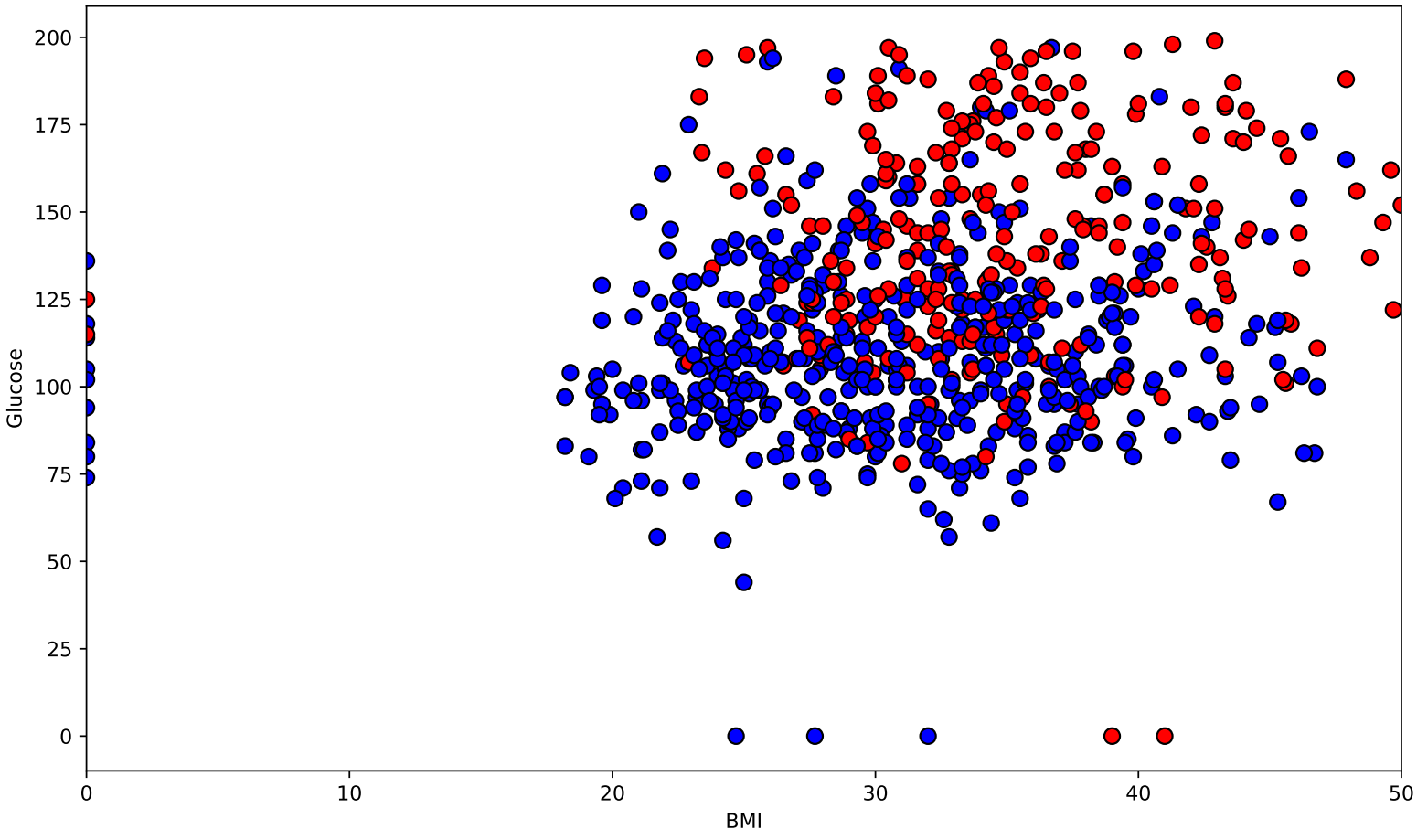

Figure: Diabetes Dataset w/ Glucose (a bit complicated)

Pseudocode

Algorithm 1: CART Algorithm [2]

```
function FIT(samples, labels, featureCount)
    if homogeneous(labels) then
        return Node(majorityClass(labels))
    end if
    bestSplit, bestSplittingScore ← None, worstSplittingScore()
    for sample in samples do
        for feature in range(0,featureCount) do
            currentSplit ← (feature, sample[feature])
            currentSplittingScore ← evaluateSplit(currentSplit, samples)
            if isBetterThan(currentSplittingScore, bestSplittingScore) then
                bestSplittingScore, bestSplit ← currentSplittingScore, currentSplit
            end if
        end for
    end for
    left, right ← splitDataByBestSplit(samples, labels, bestSplit)
    if left is empty or right is empty then
        return Node(majorityClass(labels))
    end if
    leftSubtree ← fit(left.samples, left.labels, featureCount)
    rightSubtree ← fit(right.samples, right.labels, featureCount)
    tree ← Node(bestSplit)
    tree.left, tree.right ← leftSubtree, rightSubtree
    return tree
end function
```

Quantification of Goodness (Splitting Criterion)

$$G = p_L \left(1 - \sum_{j \in C} p_{Lj}^2\right) + p_R \left(1 - \sum_{j \in C} p_{Rj}^2\right)$$

Boundary (Depth = 1)

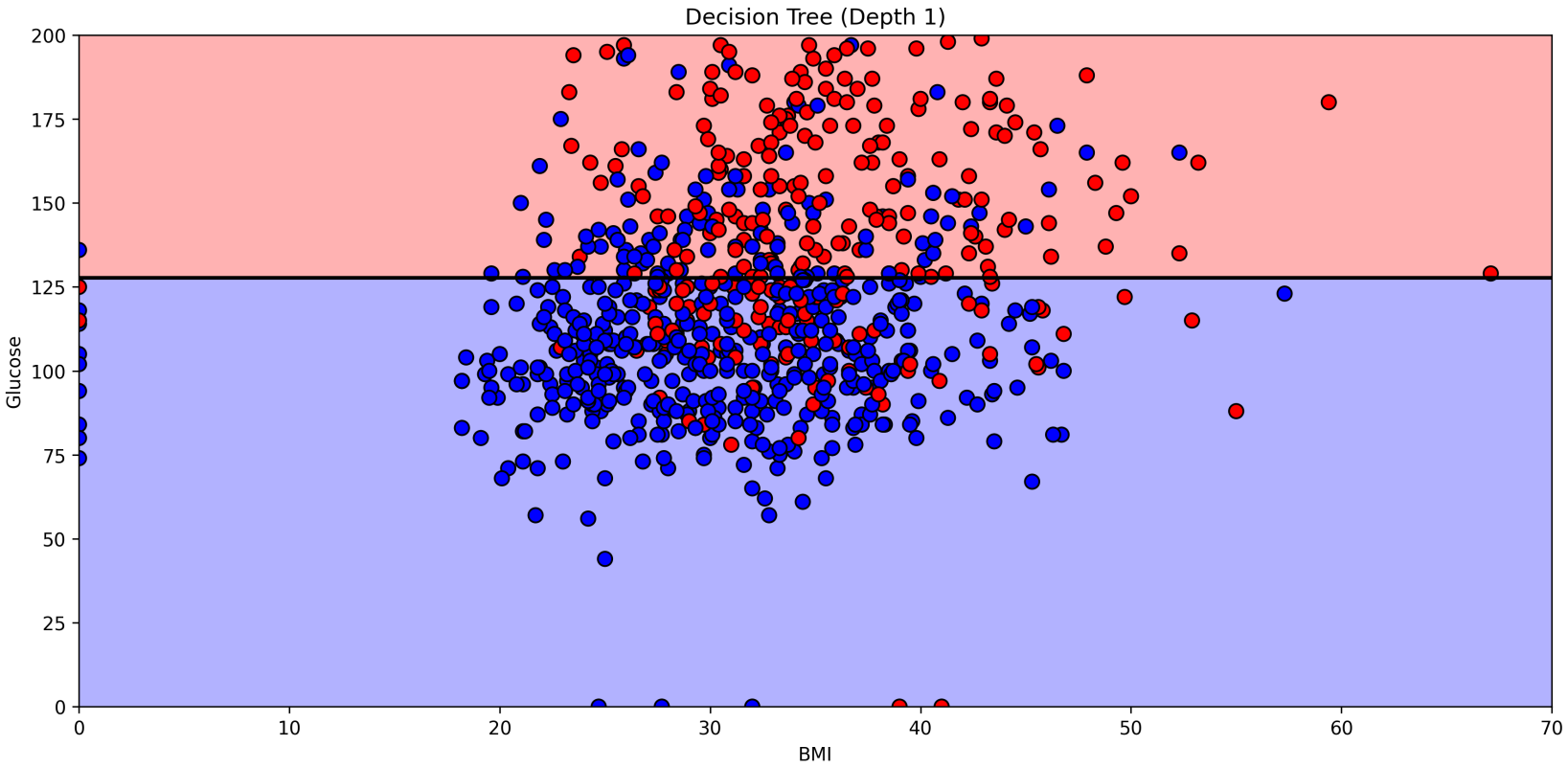

Figure: 73.6% Accuracy

Boundary (Depth = 2)

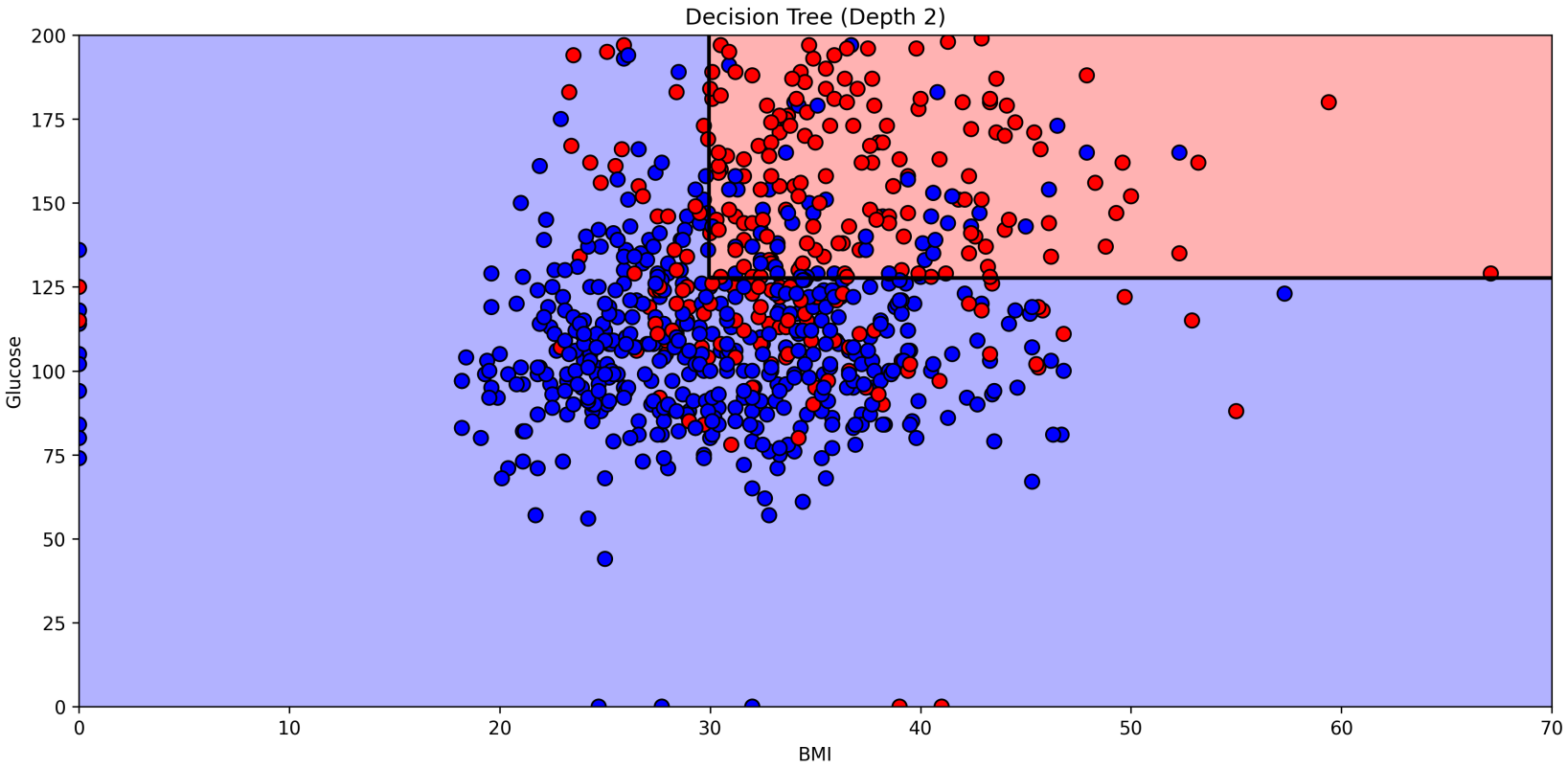

Figure: 77.2% Accuracy

Boundary (Depth = 3)

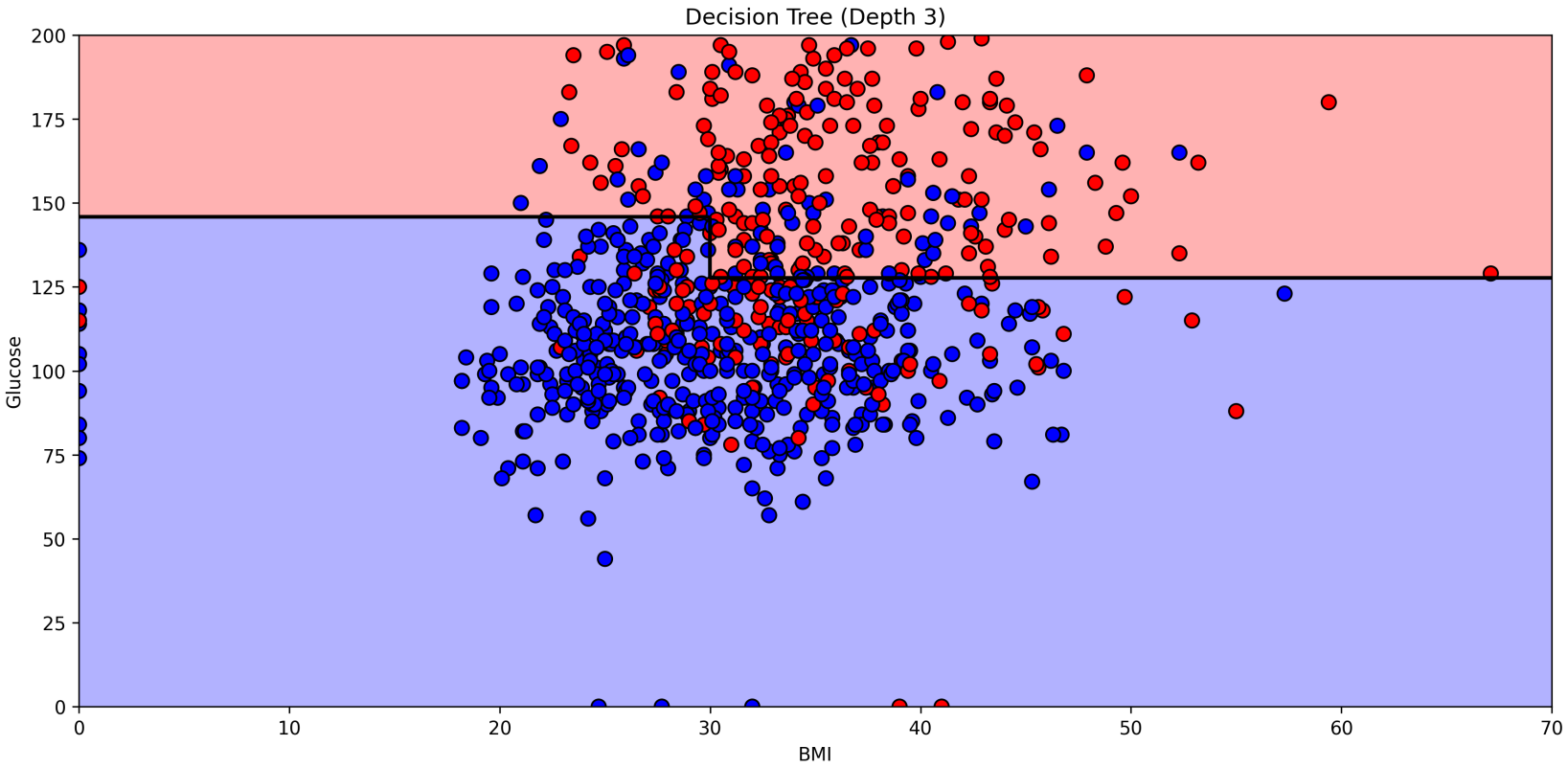

Figure: 77.3% Accuracy

Boundary (Depth = 4)

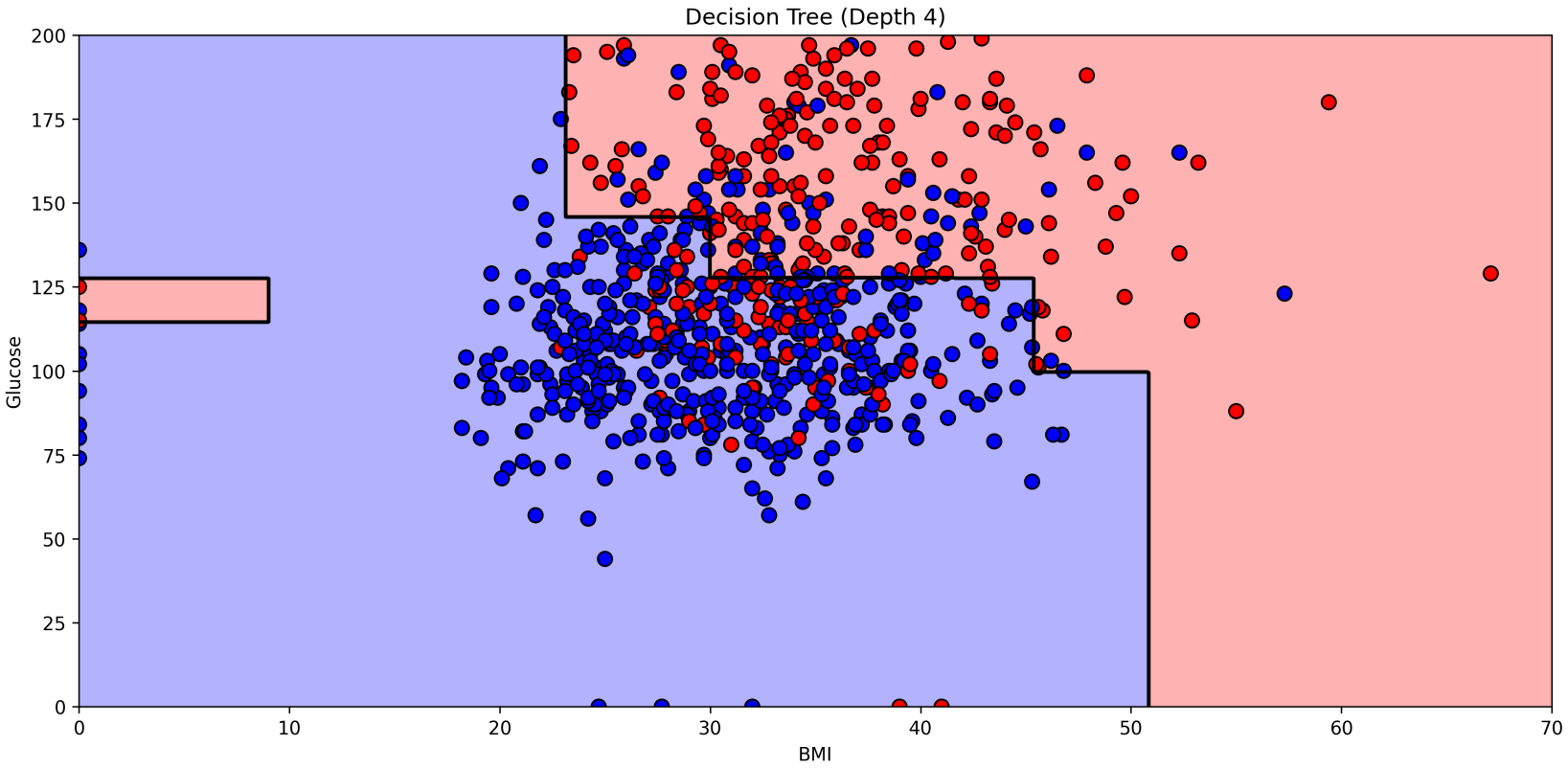

Figure: 78.5% Accuracy

Tree (Depth = 4)

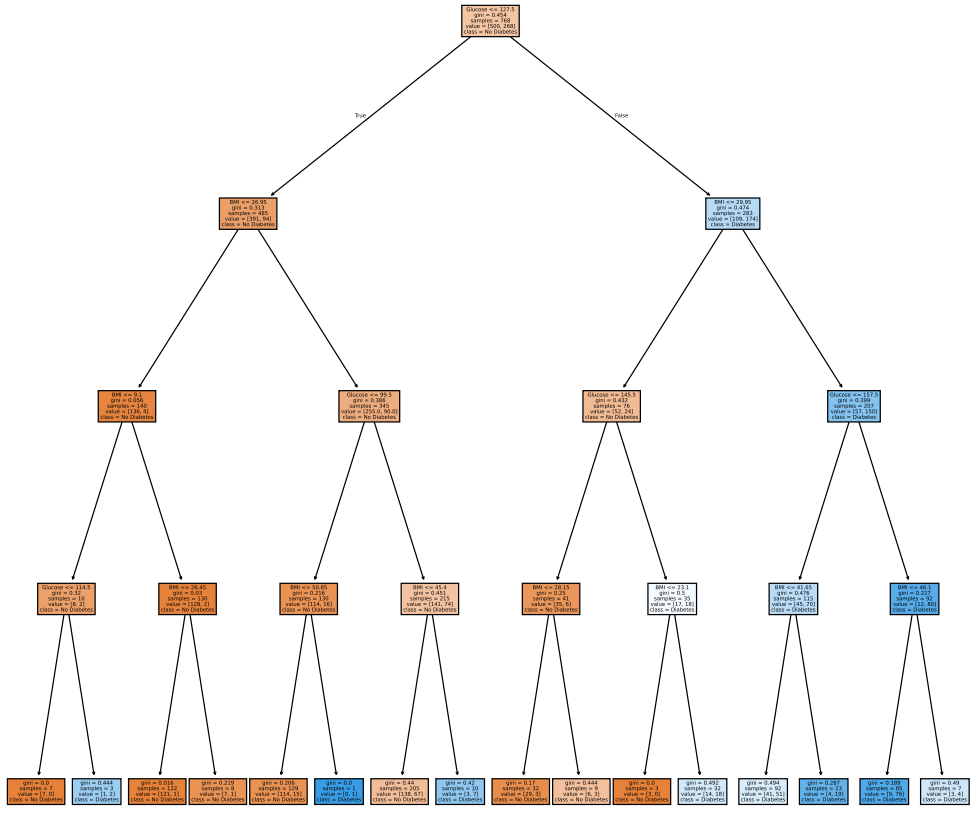

Figure: Tree Representation

New Problem

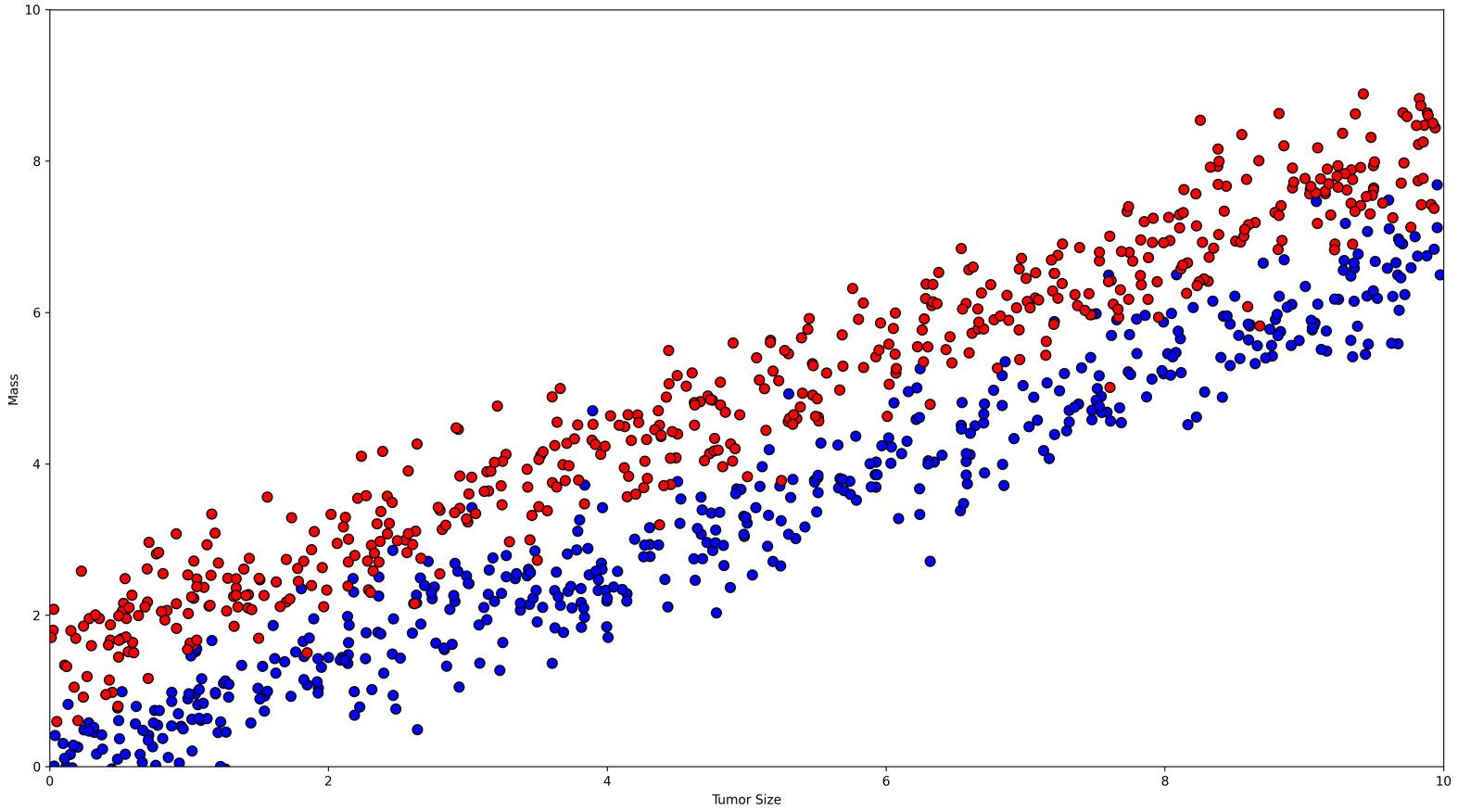

Figure: Cancer Diagnosis (Synthetic)

Boundary (Depth = 1)

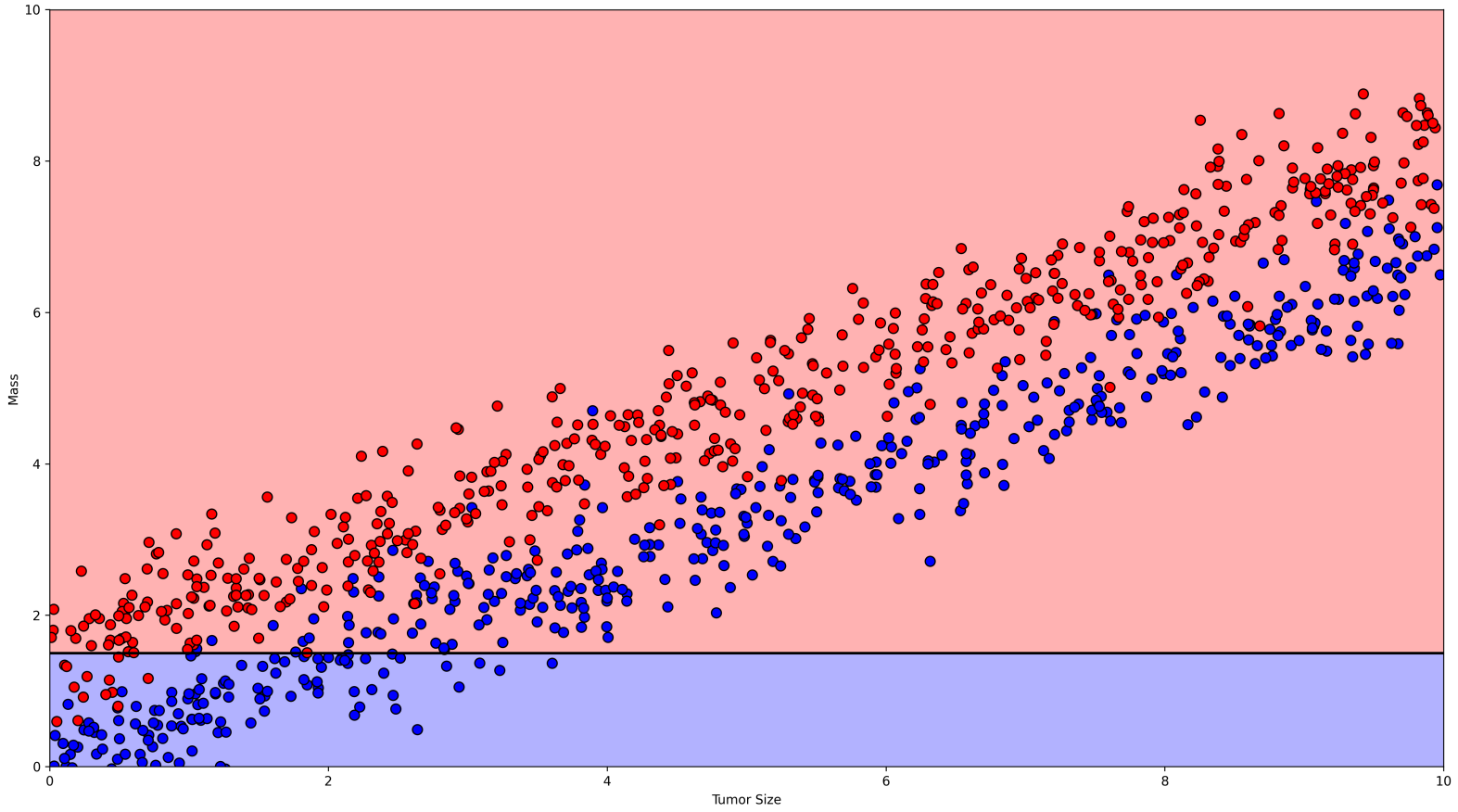

Figure: Cancer Diagnosis (Synthetic)

Boundary (Depth = 5)

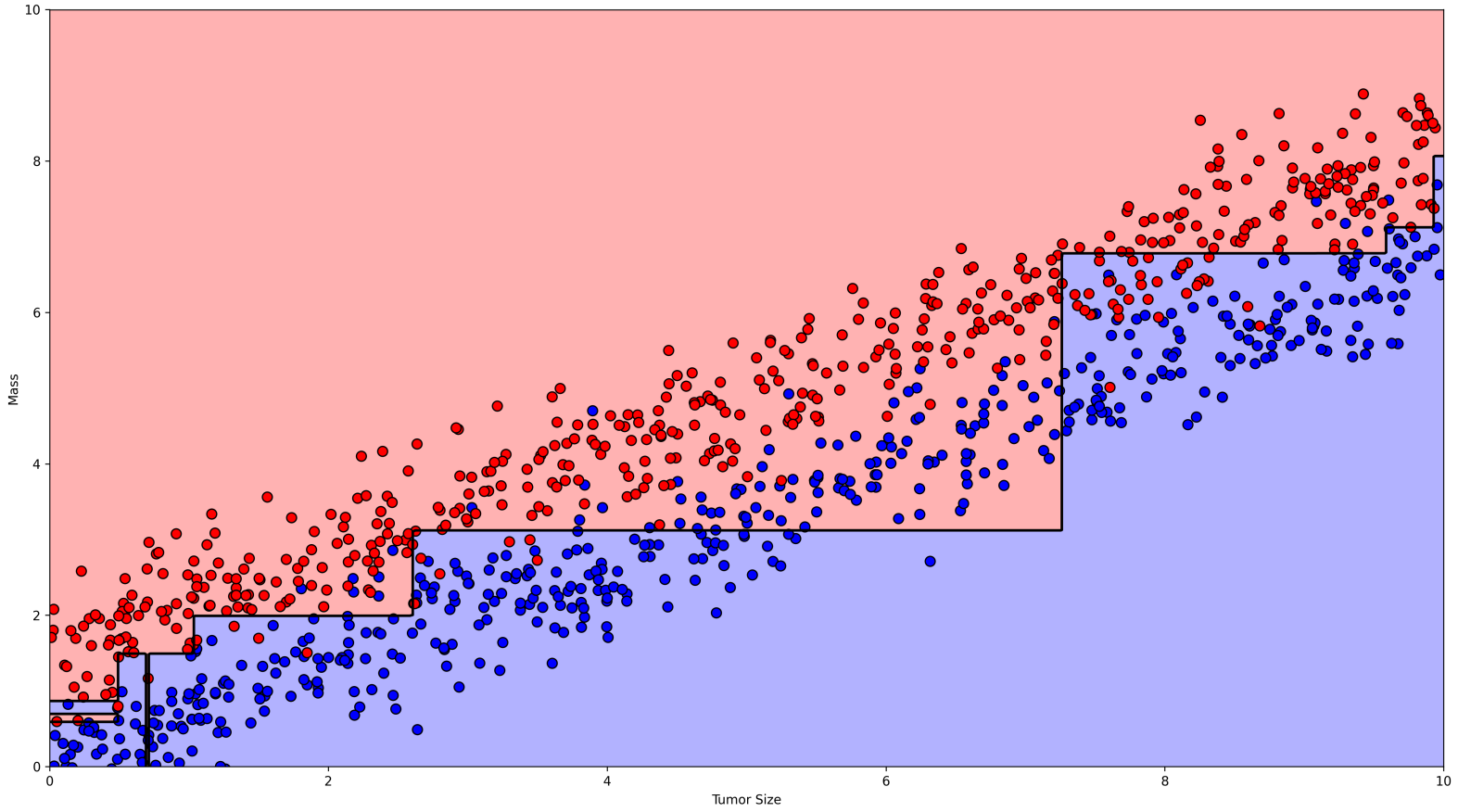

Figure: Cancer Diagnosis (Synthetic)

Boundary (Depth = 10)

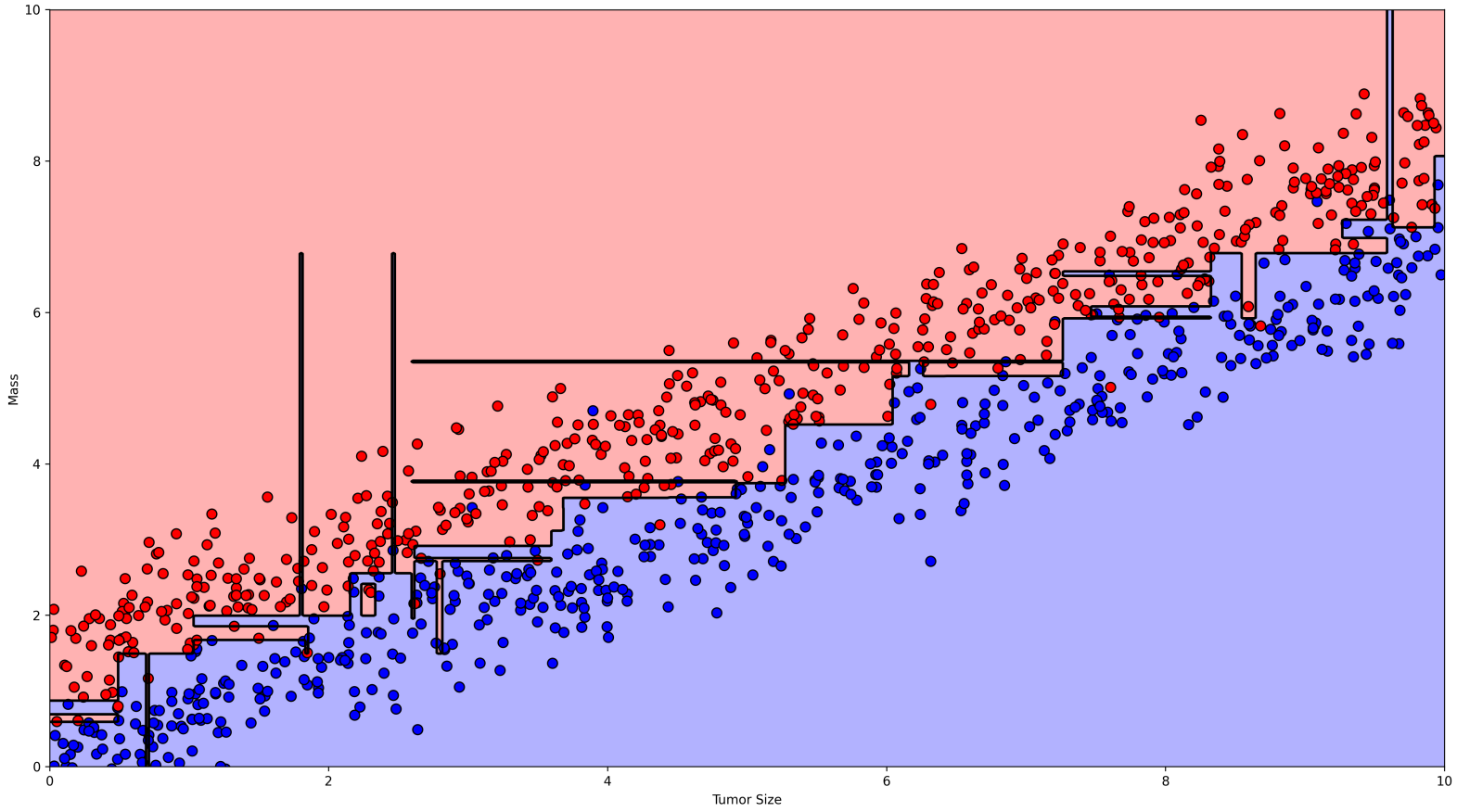

Figure: Cancer Diagnosis (Synthetic)

Boundary (Depth = 20)

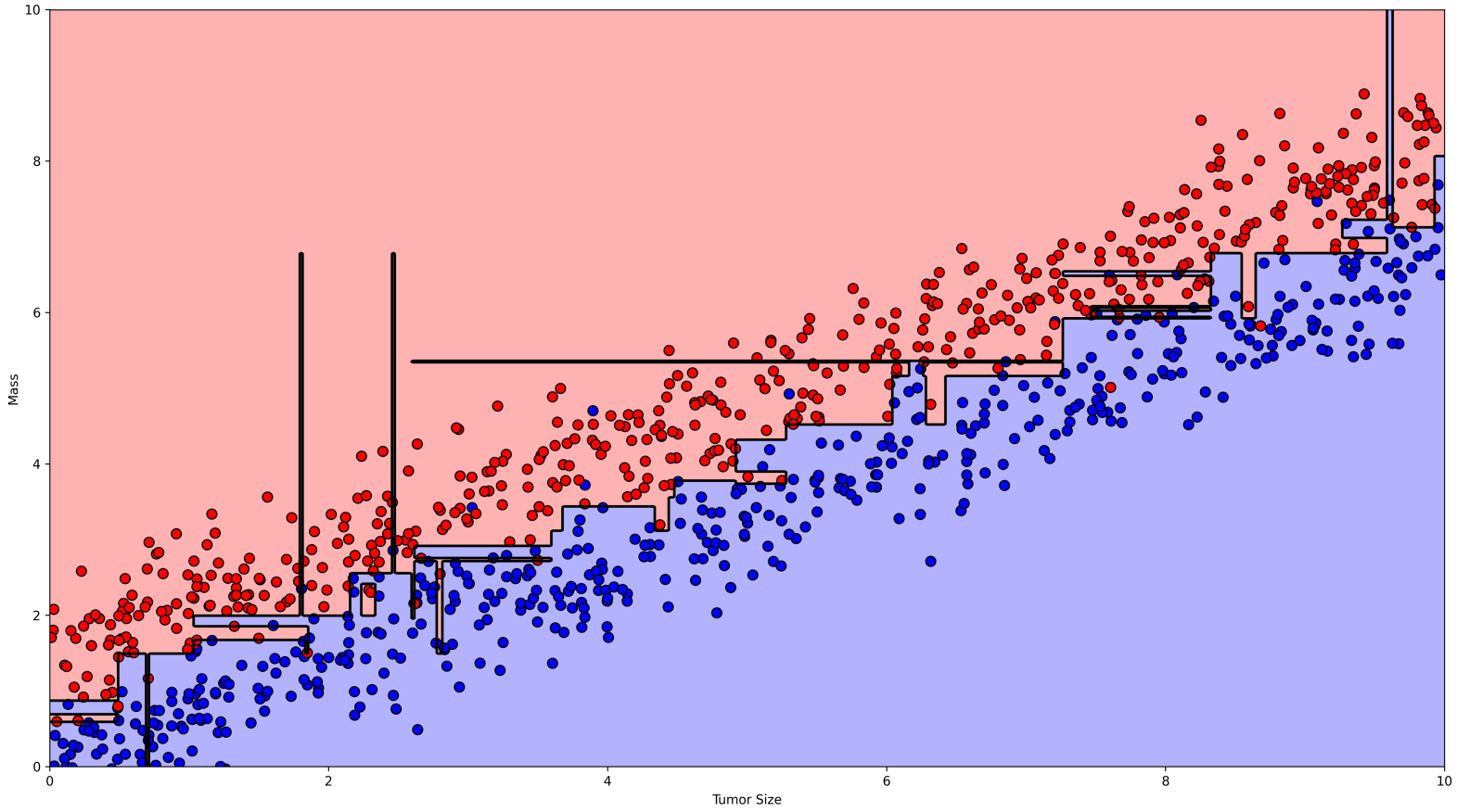

Figure: Cancer Diagnosis (Synthetic)

Tree (Depth = 20)

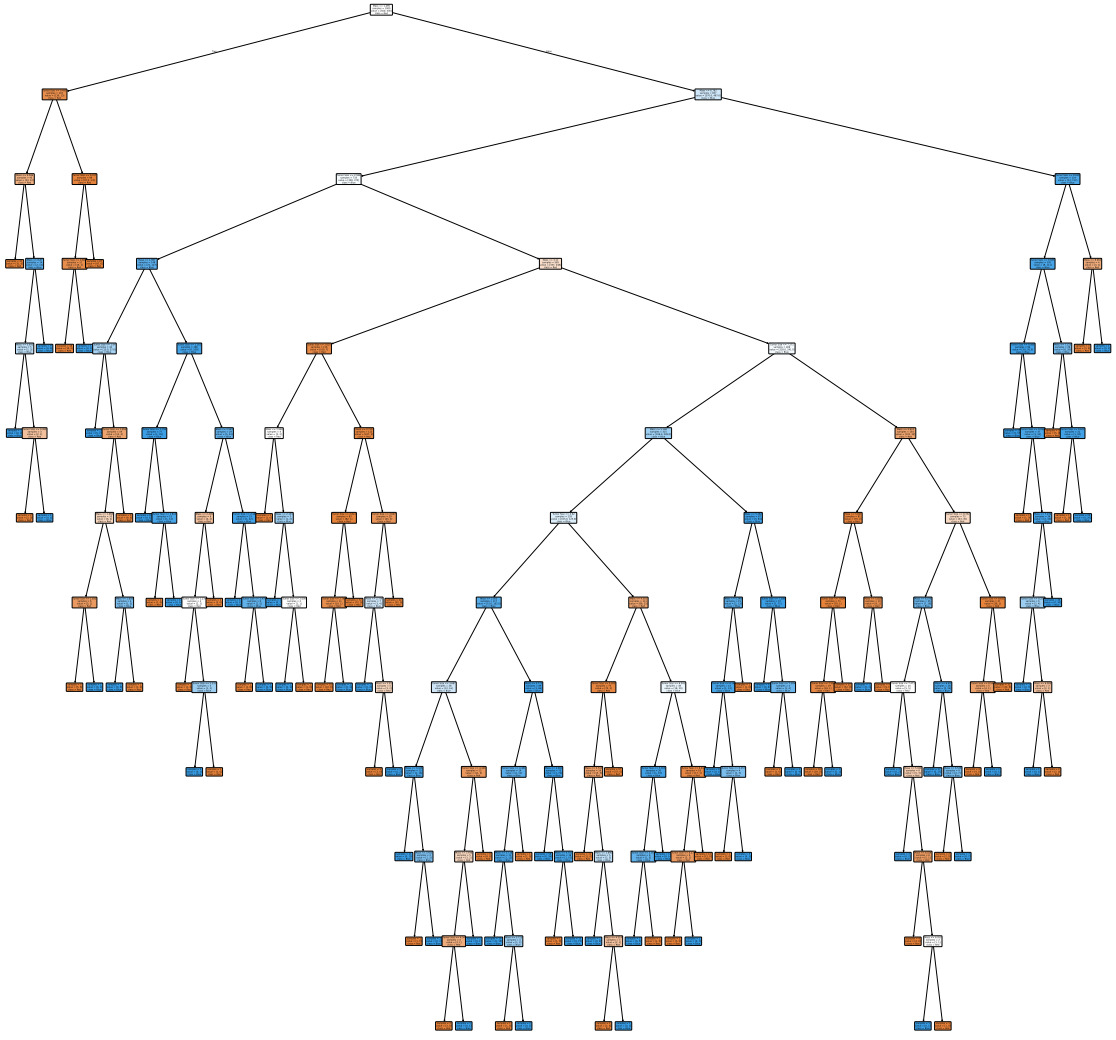

Figure: Tree Visualization

Solution

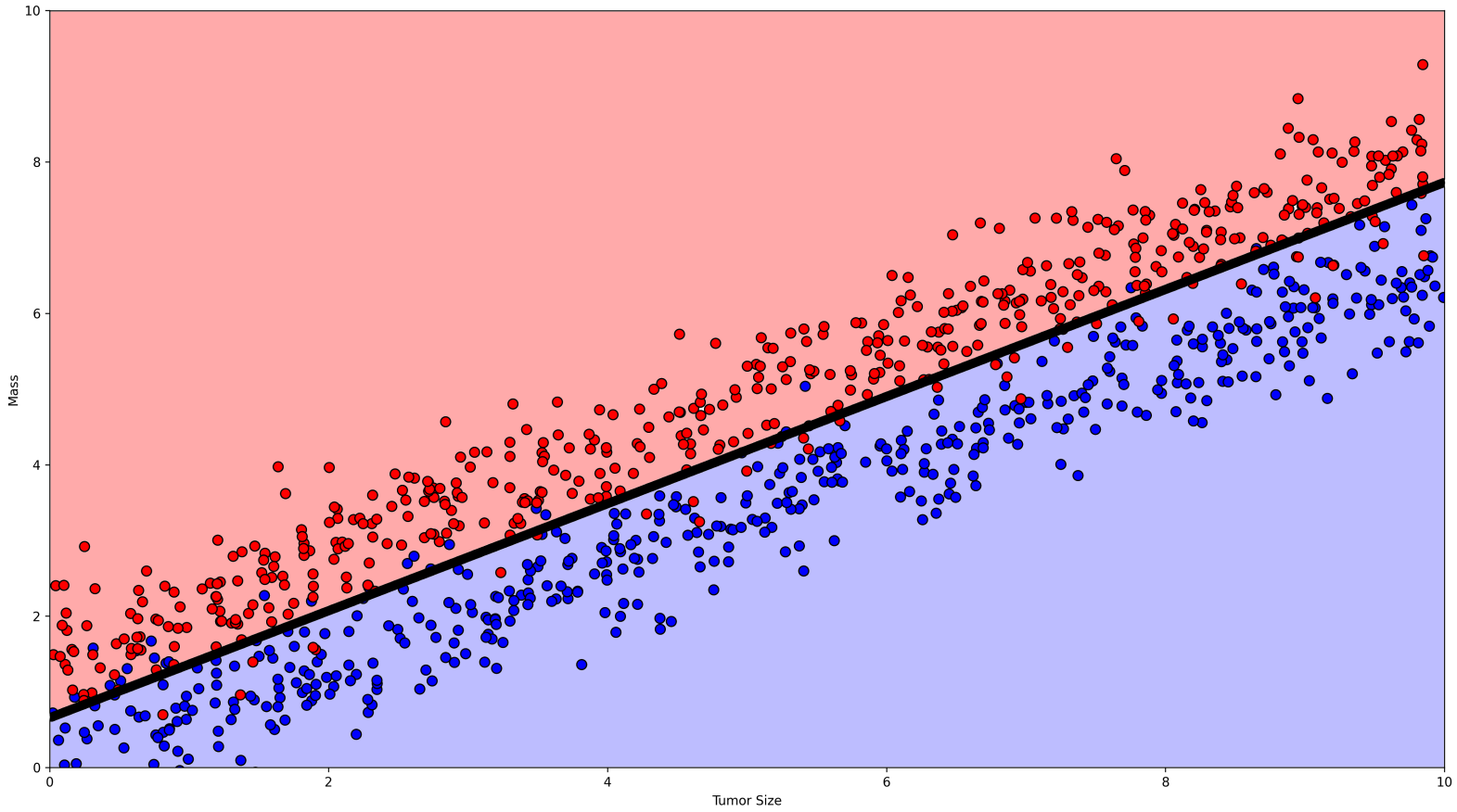

Figure: Oblique Split

How?

1. CART-LC [2]
2. OC1 [5]
3. HHCART [7]
4. CART-ELC [4]

Pseudocode

Algorithm 2: CART-ELC

```
function FIT(samples, labels, m, r)
    if homogeneous(labels) then
        return Node(majorityClass(labels))
    end if
    bestSplit, bestSplittingScore ← None, worstSplittingScore()
    for selectedSamples in combinations(samples, r) do
        for selectedFeatures in combinations(m, r) do
            vectorsToPassThrough ← featureSubset(selectedSamples, selectedFeatures)
            currentSplit ← findHyperplanePassingThrough(vectorsToPassThrough)
            currentSplittingScore ← evaluateSplit(currentSplit, samples)
            if isBetterThan(currentSplittingScore, bestSplittingScore) then
                bestSplittingScore, bestSplit ← currentSplittingScore, currentSplit
            end if
        end for
    end for
    left, right ← splitDataByBestSplit(samples, labels, bestSplit)
    if left is empty or right is empty then
        return Node(majorityClass(labels))
    end if
    leftSubtree ← fit(left.samples, left.labels, m, r)
    rightSubtree ← fit(right.samples, right.labels, m, r)
    tree ← Node(bestSplit)
    tree.left, tree.right ← leftSubtree, rightSubtree
    return tree
end function
```

Asymptotic Time Complexity per Split (CART-ELC)

$$\Theta\left(\binom{n}{r} \cdot \binom{m}{r} \cdot r(r^2 + n)\right)$$

Operations for a Single Split (CART-ELC)

Table: Operations for single split assuming multiplicative factor of one, no additive constants, and $r = m$.

r	n					
	100	500	1000	5000	10000	20000
1	1.01e+04	2.50e+05	1.00e+06	2.50e+07	1.00e+08	4.00e+08
2	1.03e+06	1.26e+08	1.00e+09	1.25e+11	1.00e+12	8.00e+12
3	5.29e+07	3.16e+10	5.03e+11	3.13e+14	5.00e+15	8.00e+16
4	1.82e+09	5.31e+12	1.68e+14	5.22e+17	1.67e+19	5.34e+20
5	4.71e+10	6.70e+14	4.23e+16	6.53e+20	4.17e+22	2.67e+24
6	9.73e+11	6.77e+16	8.50e+18	6.54e+23	8.35e+25	1.07e+28
7	1.67e+13	5.71e+18	1.43e+21	5.46e+26	1.39e+29	3.56e+31
8	2.44e+14	4.13e+20	2.05e+23	3.90e+29	1.99e+32	1.02e+35
9	3.10e+15	2.62e+22	2.59e+25	2.44e+32	2.49e+35	2.55e+38
10	3.46e+16	1.47e+24	2.90e+27	1.36e+35	2.77e+38	5.66e+41

Empirical Comparison

Table: Accuracy and tree size comparison across decision tree induction algorithms.

Algorithm	Accuracy					
	S/G Bright	S/G Dim	Cancer	Iris	Housing	Diabetes
CART-ELC	**98.9 ± 0.2**	**95.2 ± 0.5**	96.3 ± 0.4	95.1 ± 0.8	83.5 ± 0.7	**74.5 ± 1.3**
HHCART(A)	98.3 ± 0.5	93.7 ± 0.8	**96.9 ± 0.3**	**95.5 ± 1.4**	**83.9 ± 0.8**	73.2 ± 1.2
HHCART(D)	98.1 ± 0.4	93.7 ± 0.9	**96.9 ± 0.3**	94.3 ± 1.5	82.2 ± 1.4	73.2 ± 1.2
OC1	**98.9 ± 0.2**	95.0 ± 0.3	96.2 ± 0.3	94.7 ± 3.1	82.4 ± 0.8	74.4 ± 1.0
OC1-AP	98.1 ± 0.2	94.0 ± 0.2	94.5 ± 0.5	92.7 ± 2.4	81.8 ± 1.0	73.8 ± 1.0
CART-LC	98.8 ± 0.2	92.8 ± 0.5	95.3 ± 0.6	93.5 ± 2.9	81.4 ± 1.2	73.7 ± 1.2
CART-AP	98.5 ± 0.5	94.2 ± 0.7	95.0 ± 1.6	93.8 ± 3.7	82.1 ± 3.5	73.9 ± 3.4
C4.5	98.5 ± 0.5	93.3 ± 0.8	95.3 ± 2.0	95.1 ± 3.2	83.2 ± 3.1	71.4 ± 3.3
Algorithm	Tree Size					
	S/G Bright	S/G Dim	Cancer	Iris	Housing	Diabetes
CART-ELC	**3.7 ± 0.2**	**9.8 ± 4.2**	**2.0 ± 0.0**	4.8 ± 0.1	**4.0 ± 0.0**	**4.0 ± 0.0**
HHCART(A)	6.1 ± 0.3	14.6 ± 4.8	**2.0 ± 0.0**	**3.1 ± 0.1**	7.8 ± 0.2	**4.0 ± 0.0**
HHCART(D)	6.3 ± 0.4	14.9 ± 5.0	**2.0 ± 0.0**	4.7 ± 0.1	23.3 ± 0.8	**4.0 ± 0.0**
OC1	4.3 ± 1.0	13.0 ± 8.7	2.8 ± 0.9	**3.1 ± 0.2**	6.9 ± 3.2	5.4 ± 3.8
OC1-AP	6.9 ± 2.4	29.3 ± 8.8	6.4 ± 1.7	3.2 ± 0.3	8.6 ± 4.5	11.4 ± 7.5
CART-LC	3.9 ± 1.3	24.2 ± 8.7	3.5 ± 0.9	3.2 ± 0.3	5.8 ± 3.2	8.0 ± 5.2
CART-AP	13.9 ± 5.7	30.4 ± 10.0	11.5 ± 7.2	4.3 ± 1.6	15.1 ± 10	11.5 ± 9.1
C4.5	14.3 ± 2.2	77.9 ± 7.4	9.8 ± 2.2	4.6 ± 0.8	28.2 ± 3.3	56.3 ± 7.9

Cohen's d

Table: Cohen's d effect size for accuracies between various models and CART-ELC. Comparisons with $p < 0.05$ are bolded.

Algorithm	S/G Bright	S/G Dim	Cancer	Iris	Housing	Diabetes
HHCART(A)	**1.576**	**2.249**	**-1.697**	-0.351	-0.532	**1.039**
HHCART(D)	**2.530**	**2.060**	**-1.697**	0.666	**1.175**	**1.039**
OC1	0.000	0.485	0.283	0.177	**1.463**	0.086
CART-LC	0.500	**4.800**	**1.961**	0.752	**2.138**	0.639
OC1-AP	**4.000**	**3.151**	**3.976**	**1.342**	**1.970**	0.604
CART-AP	**1.050**	**1.644**	**1.115**	0.486	0.555	0.233
C4.5	**1.050**	**2.848**	0.693	0.000	0.133	**1.236**

Future Directions

1. Smaller Subsets of Candidates
   1.1 Active Sampling
   1.2 Feature Selection
2. Random Forests [1]
3. Stochastic Gradient Boosting [3]

References

[1] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, Oct 2001.

[2] L. Breiman, J. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Chapman and Hall/CRC, 1st edition, 1984.

[3] J. H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378, 2002.

[4] A. D. Laack. Cart-elc: Oblique decision tree induction via exhaustive search. Manuscript under review at Transactions on Machine Learning Research (TMLR), 2025.

[5] S. K. Murthy, S. Kasif, and S. L. Salzberg. A system for induction of oblique decision trees. *Journal of Artificial Intelligence Research*, 2:1–32, 1994.

[6] J. W. Smith, J. E. Everhart, W. C. Dickson, W. C. Knowler, and R. S. Johannes. Using the adap learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*, pages 261–265, 1988.

[7] D. C. Wickramarachchi, B. L. Robertson, M. Reale, C. J. Price, and J. Brown. HHCART: An oblique decision tree. *Computational Statistics & Data Analysis*, 96:12–23, 2016.