# Modal Symbolic Learning: Day 3

## Symbolic Land Cover Classification (*unimodal data*, *multimodal learning*)

In [1]:
```julia
using Pkg
Pkg.activate(".")
Pkg.instantiate()
Pkg.update()
Pkg.status()
```

```
  Activating project at `~/Desktop/modal-symbolic-learning-course`
    Updating registry at `~/.julia/registries/General`
    Updating git-repo `https://github.com/JuliaRegistries/General.git`
  No Changes to `~/Desktop/modal-symbolic-learning-course/Project.toml`
  No Changes to `~/Desktop/modal-symbolic-learning-course/Manifest.toml`
Status `~/Desktop/modal-symbolic-learning-course/Project.toml`
  [acdeb78f] Catch22 v0.4.5
  [a93c6f00] DataFrames v1.6.1
  [864edb3b] DataStructures v0.18.15
  [7806a523] DecisionTree v0.12.4
  [7073ff75] IJulia v1.24.2
  [6a3955dd] ImageFiltering v0.7.8
  [033835bb] JLD2 v0.4.38
  [23992714] MAT v0.10.6
⌅ [add582a8] MLJ v0.19.5
  [c6f25543] MLJDecisionTreeInterface v0.4.0
  [e54bda2e] ModalDecisionTrees v0.3.6
  [91a5bcdd] Plots v1.39.0
  [7b3b3b3f] Sole v0.3.1
  [b002da8f] SoleLogics v0.6.14
  [4249d9c7] SoleModels v0.5.6
  [2913bbd2] StatsBase v0.34.2
  [9a3f8284] Random
Info Packages marked with ⌅ have new versions available but compatibility co
nstraints restrict them from upgrading. To see why use `status --outdated`
```

In [2]:
```julia
# Import libraries for statistics & Machine Learning
using Random
using DataFrames
using Plots
using StatsBase

using MLJ
using Sole
```

In [3]:
```julia
include("land-cover.jl")
data_dir = "data/"

X_df, y = LandCoverDataset(
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

```julia
    "Pavia University";
    window_size          = 3,
    ninstances_per_class = 40,
    pad_window_size      = 5,
);
```

```
Load LandCoverDataset: Pavia University...
window_size         = (3, 3)
pad_window_size     = (5, 5)
ninstances_per_class  = 40
ninstances_per_class_strategy = updownsampling
flattened           = false
apply_filter        = false
seed                = 1

Image size: (610, 340, 103)
class_counts_d = [("Asphalt", 1 => 6631), ("Meadows", 2 => 18649), ("Grave
l", 3 => 2099), ("Trees", 4 => 3064), ("Painted metal sheets", 5 => 1345),
("Bare Soil", 6 => 5029), ("Bitumen", 7 => 1330), ("Self-Blocking Bricks", 8
=> 3682), ("Shadows", 9 => 947)]
no_class_counts = 164624
n_classes = 9
ninstances = 40 * 9 = 360
effective_class_counts_d = [("Asphalt", 1 => 40), ("Meadows", 2 => 40), ("Gr
avel", 3 => 40), ("Trees", 4 => 40), ("Painted metal sheets", 5 => 40), ("Ba
re Soil", 6 => 40), ("Bitumen", 7 => 40), ("Self-Blocking Bricks", 8 => 40),
("Shadows", 9 => 40)]
countmap(labels) = Dict(5 => 40, 4 => 40, 6 => 40, 7 => 40, 2 => 40, 9 => 4
0, 8 => 40, 3 => 40, 1 => 40)
```

In [4]:

```julia
countmap(y)
```

Out[4]:

```
Dict{String, Int64} with 9 entries:
  "Self-Blocking Bricks" => 40
  "Bitumen"              => 40
  "Gravel"               => 40
  "Bare Soil"            => 40
  "Painted metal sheets" => 40
  "Shadows"              => 40
  "Trees"                => 40
  "Asphalt"              => 40
  "Meadows"              => 40
```

In [5]:

```julia
length.(X_df)
```

Out[5]: 360×103 DataFrame *3 columns and 335 rows omitted*

| Row | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 | V9 | V10 | V1 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Int64 | Int64 | Int64 | Int64 | Int64 | Int64 | Int64 | Int64 | Int64 | Int64 | In |
| **1** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **2** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **3** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **4** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **5** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **6** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **7** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **8** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **9** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **10** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **11** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **12** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **13** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | |
| **349** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **350** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **351** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **352** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **353** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **354** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **355** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **356** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **357** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **358** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **359** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |
| **360** | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | |

In [49]:
```
X_df = broadcast(values->Matrix{Float64}(values), X_df)
```

Out[49]: 360×103 DataFrame *3 columns and 335 rows omitted*

| Row | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 |
|---|---|---|---|---|---|---|---|---|
| | **Array…** | **Array…** | **Array…** | **Array…** | **Array…** | **Array…** | **Array…** | **Array…** |
| **1** | [1186.0 1294.0 1291.0; 1701.0 1705.0 1373.0; 1823.0 1359.0 1616.0] | [1069.0 1177.0 1369.0; 1708.0 1463.0 1041.0; 2055.0 1229.0 1440.0] | [1176.0 1179.0 1265.0; 1686.0 1190.0 939.0; 1967.0 1256.0 1245.0] | [1318.0 1256.0 1273.0; 1528.0 1106.0 1133.0; 1829.0 1267.0 1085.0] | [1278.0 1286.0 1326.0; 1414.0 1082.0 1308.0; 1759.0 1231.0 1031.0] | [1183.0 1290.0 1254.0; 1436.0 1130.0 1411.0; 1740.0 1150.0 1087.0] | [1120.0 1238.0 1177.0; 1423.0 1268.0 1329.0; 1731.0 1165.0 1165.0] | [1199.0 1184.0 1084.0; 1346.0 1293.0 1228.0; 1745.0 1214.0 1167.0] |
| **2** | [1153.0 1201.0 1563.0; 784.0 1618.0 1508.0; 1173.0 1071.0 1434.0] | [1024.0 1258.0 1480.0; 1024.0 1548.0 1368.0; 1110.0 1152.0 1193.0] | [1077.0 1224.0 1397.0; 1147.0 1320.0 1266.0; 1143.0 1276.0 1164.0] | [1130.0 1124.0 1341.0; 1165.0 1220.0 1282.0; 1233.0 1326.0 1345.0] | [1245.0 1214.0 1330.0; 1122.0 1311.0 1406.0; 1329.0 1316.0 1519.0] | [1325.0 1287.0 1352.0; 1163.0 1357.0 1463.0; 1368.0 1309.0 1630.0] | [1388.0 1351.0 1362.0; 1184.0 1391.0 1468.0; 1346.0 1323.0 1645.0] | [1342.0 1401.0 1351.0; 1146.0 1392.0 1367.0; 1268.0 1324.0 1563.0] |
| **3** | [880.0 804.0 727.0; 672.0 865.0 1067.0; 540.0 757.0 803.0] | [726.0 486.0 771.0; 498.0 629.0 957.0; 676.0 886.0 817.0] | [561.0 509.0 648.0; 401.0 548.0 808.0; 791.0 784.0 733.0] | [579.0 535.0 759.0; 536.0 768.0 714.0; 769.0 773.0 669.0] | [623.0 651.0 684.0; 608.0 809.0 736.0; 722.0 758.0 729.0] | [690.0 661.0 551.0; 570.0 701.0 763.0; 740.0 654.0 694.0] | [688.0 627.0 530.0; 615.0 613.0 713.0; 765.0 602.0 722.0] | [644.0 670.0 552.0; 658.0 663.0 684.0; 729.0 705.0 700.0] |
| **4** | [1454.0 1082.0 1291.0; 512.0 985.0 936.0; 1430.0 1400.0 1098.0] | [1178.0 876.0 1229.0; 741.0 1014.0 1064.0; 1124.0 1259.0 1282.0] | [952.0 881.0 1201.0; 1074.0 901.0 1014.0; 1149.0 1249.0 1286.0] | [1071.0 1034.0 1198.0; 1251.0 1039.0 996.0; 1238.0 1215.0 1107.0] | [1162.0 1067.0 1305.0; 1243.0 1143.0 996.0; 1206.0 1151.0 1054.0] | [988.0 1072.0 1286.0; 1167.0 1080.0 1019.0; 1119.0 1053.0 1019.0] | [921.0 1049.0 1130.0; 1161.0 1078.0 1061.0; 1075.0 1047.0 1030.0] | [958.0 958.0 1087.0; 1158.0 1004.0 1066.0; 1098.0 1095.0 1084.0] |
| **5** | [1603.0 1091.0 594.0; 1003.0 1260.0 891.0; 1222.0 801.0 863.0] | [1602.0 1092.0 689.0; 804.0 1240.0 827.0; 1002.0 941.0 736.0] | [1427.0 1149.0 847.0; 790.0 1191.0 905.0; 1081.0 1126.0 863.0] | [1308.0 1144.0 893.0; 901.0 1164.0 988.0; 1065.0 1273.0 1002.0] | [1354.0 1048.0 1008.0; 976.0 1227.0 978.0; 951.0 1365.0 1096.0] | [1451.0 1027.0 957.0; 993.0 1390.0 909.0; 765.0 1500.0 1110.0] | [1545.0 1032.0 842.0; 977.0 1572.0 916.0; 674.0 1600.0 1073.0] | [1536.0 1057.0 770.0; 954.0 1578.0 949.0; 733.0 1553.0 1034.0] |
| **6** | [966.0 632.0 605.0; 482.0 669.0 511.0; 427.0 | [1171.0 630.0 451.0; 446.0 507.0 471.0; 425.0 | [999.0 505.0 521.0; 413.0 683.0 462.0; 489.0 | [790.0 450.0 483.0; 359.0 646.0 494.0; 546.0 | [632.0 420.0 468.0; 350.0 564.0 476.0; 559.0 | [478.0 470.0 549.0; 446.0 436.0 439.0; 542.0 | [523.0 551.0 568.0; 532.0 418.0 410.0; 540.0 | [540.0 511.0 471.0; 473.0 480.0 459.0; 511.0 |

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

| Row | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 |
|---|---|---|---|---|---|---|---|---|
| | Array… | Array… | Array… | Array… | Array… | Array… | Array… | Array… |
| | 811.0 640.0] | 438.0 687.0] | 191.0 701.0] | 360.0 583.0] | 510.0 436.0] | 514.0 326.0] | 532.0 350.0] | 484.0 377.0] |
| 7 | [1219.0 1459.0 1247.0; 1377.0 483.0 1501.0; 1187.0 773.0 1630.0] | [1377.0 1259.0 1255.0; 1298.0 885.0 1456.0; 1380.0 975.0 1521.0] | [1391.0 1225.0 1282.0; 1223.0 1091.0 1404.0; 1282.0 1141.0 1370.0] | [1359.0 1340.0 1325.0; 1292.0 1260.0 1356.0; 1327.0 1167.0 1450.0] | [1411.0 1436.0 1401.0; 1366.0 1314.0 1262.0; 1390.0 1176.0 1459.0] | [1388.0 1450.0 1427.0; 1446.0 1241.0 1351.0; 1370.0 1172.0 1384.0] | [1306.0 1389.0 1447.0; 1398.0 1270.0 1468.0; 1301.0 1245.0 1415.0] | [1215.0 1377.0 1525.0; 1306.0 1261.0 1383.0; 1325.0 1333.0 1429.0] |
| 8 | [510.0 686.0 863.0; 637.0 748.0 988.0; 815.0 686.0 872.0] | [365.0 726.0 761.0; 810.0 639.0 713.0; 875.0 542.0 782.0] | [207.0 602.0 854.0; 714.0 523.0 611.0; 711.0 440.0 681.0] | [445.0 341.0 833.0; 663.0 670.0 695.0; 616.0 445.0 620.0] | [636.0 302.0 736.0; 606.0 728.0 750.0; 579.0 454.0 553.0] | [689.0 498.0 704.0; 503.0 583.0 737.0; 553.0 483.0 504.0] | [634.0 629.0 646.0; 485.0 548.0 683.0; 602.0 573.0 508.0] | [646.0 609.0 599.0; 519.0 557.0 619.0; 592.0 574.0 577.0] |
| 9 | [541.0 814.0 1198.0; 857.0 1166.0 812.0; 1317.0 620.0 1044.0] | [530.0 676.0 910.0; 697.0 848.0 546.0; 1170.0 713.0 905.0] | [546.0 625.0 708.0; 709.0 541.0 654.0; 1063.0 754.0 778.0] | [718.0 702.0 791.0; 759.0 646.0 872.0; 861.0 592.0 646.0] | [733.0 763.0 744.0; 813.0 759.0 851.0; 799.0 669.0 595.0] | [783.0 798.0 638.0; 876.0 716.0 747.0; 750.0 861.0 656.0] | [834.0 857.0 664.0; 836.0 730.0 783.0; 699.0 794.0 747.0] | [762.0 838.0 793.0; 781.0 803.0 819.0; 649.0 727.0 794.0] |
| 10 | [856.0 302.0 568.0; 463.0 460.0 870.0; 631.0 762.0 801.0] | [698.0 291.0 598.0; 144.0 563.0 822.0; 494.0 484.0 525.0] | [367.0 449.0 460.0; 59.0 527.0 724.0; 292.0 146.0 293.0] | [264.0 449.0 346.0; 389.0 444.0 586.0; 211.0 183.0 287.0] | [351.0 349.0 209.0; 513.0 392.0 399.0; 179.0 255.0 352.0] | [379.0 364.0 256.0; 421.0 311.0 245.0; 117.0 351.0 304.0] | [304.0 380.0 335.0; 333.0 284.0 263.0; 133.0 334.0 258.0] | [257.0 359.0 331.0; 342.0 311.0 341.0; 195.0 312.0 284.0] |
| 11 | [589.0 838.0 602.0; 1055.0 877.0 1041.0; 747.0 1016.0 833.0] | [671.0 617.0 508.0; 800.0 688.0 749.0; 659.0 839.0 655.0] | [631.0 648.0 605.0; 715.0 711.0 391.0; 700.0 772.0 571.0] | [614.0 703.0 659.0; 763.0 682.0 381.0; 663.0 925.0 601.0] | [654.0 751.0 614.0; 821.0 701.0 522.0; 643.0 898.0 619.0] | [725.0 666.0 642.0; 841.0 728.0 674.0; 646.0 744.0 614.0] | [764.0 581.0 634.0; 845.0 717.0 694.0; 669.0 653.0 536.0] | [771.0 595.0 635.0; 735.0 636.0 611.0; 720.0 612.0 505.0] |
| 12 | [967.0 1086.0 1269.0; 1147.0 1269.0 1194.0; 907.0 | [1125.0 1005.0 1265.0; 1267.0 1122.0 1452.0; 992.0 | [1173.0 1057.0 1270.0; 1250.0 1086.0 1417.0; 1309.0 | [1176.0 1148.0 1262.0; 1202.0 1128.0 1290.0; 1347.0 | [1226.0 1260.0 1348.0; 1240.0 1248.0 1195.0; 1227.0 | [1365.0 1379.0 1455.0; 1292.0 1317.0 1212.0; 1298.0 | [1408.0 1479.0 1450.0; 1309.0 1368.0 1353.0; 1376.0 | [1338.0 1430.0 1394.0; 1323.0 1398.0 1389.0; 1424.0 |

| Row | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 |
|---|---|---|---|---|---|---|---|---|
| | Array… | Array… | Array… | Array… | Array… | Array… | Array… | Array… |
| | 1052.0 1016.0] | 988.0 1412.0] | 1160.0 1498.0] | 1202.0 1523.0] | 1145.0 1541.0] | 1213.0 1448.0] | 1324.0 1280.0] | 1386.0 1289.0] |
| **13** | [663.0 1111.0 382.0; 627.0 642.0 816.0; 893.0 474.0 438.0] | [560.0 940.0 488.0; 488.0 548.0 802.0; 1093.0 577.0 388.0] | [585.0 621.0 578.0; 532.0 279.0 575.0; 679.0 767.0 569.0] | [561.0 362.0 652.0; 680.0 407.0 523.0; 338.0 710.0 578.0] | [582.0 495.0 692.0; 684.0 548.0 653.0; 446.0 567.0 600.0] | [528.0 551.0 703.0; 547.0 493.0 699.0; 501.0 485.0 638.0] | [511.0 498.0 688.0; 510.0 529.0 692.0; 589.0 522.0 666.0] | [578.0 549.0 622.0; 513.0 581.0 669.0; 625.0 567.0 549.0] |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| **349** | [2669.0 3073.0 2410.0; 2567.0 2651.0 2144.0; 2279.0 2681.0 2580.0] | [2734.0 3217.0 2570.0; 2533.0 2546.0 2403.0; 2360.0 2665.0 2600.0] | [3207.0 3370.0 2622.0; 2766.0 2217.0 2519.0; 2535.0 2891.0 2686.0] | [3649.0 3523.0 2791.0; 3125.0 2250.0 2601.0; 2859.0 3041.0 2833.0] | [3930.0 3777.0 3039.0; 3342.0 2717.0 2715.0; 3239.0 3190.0 3103.0] | [4067.0 3974.0 3113.0; 3453.0 2892.0 2862.0; 3496.0 3481.0 3303.0] | [4144.0 4108.0 3073.0; 3628.0 2941.0 2944.0; 3551.0 3603.0 3337.0] | [4241.0 4160.0 3094.0; 3890.0 3024.0 3015.0; 3650.0 3655.0 3382.0] |
| **350** | [3427.0 3340.0 2515.0; 2280.0 2584.0 1827.0; 2478.0 2516.0 2125.0] | [3685.0 3299.0 2416.0; 2496.0 2485.0 1910.0; 2369.0 2573.0 2141.0] | [4010.0 3260.0 2516.0; 2673.0 2348.0 2059.0; 2289.0 2370.0 2247.0] | [4351.0 3458.0 2762.0; 2769.0 2518.0 2358.0; 2494.0 2272.0 2393.0] | [4440.0 3801.0 2806.0; 2804.0 2681.0 2659.0; 2674.0 2380.0 2478.0] | [4583.0 3992.0 2912.0; 2978.0 2772.0 2793.0; 2804.0 2609.0 2601.0] | [4698.0 4085.0 3072.0; 3185.0 2815.0 2771.0; 2839.0 2818.0 2730.0] | [4861.0 4100.0 3176.0; 3290.0 2791.0 2720.0; 2897.0 2882.0 2777.0] |
| **351** | [1300.0 1477.0 1108.0; 969.0 1080.0 1134.0; 1005.0 591.0 867.0] | [1289.0 1252.0 1139.0; 1172.0 1001.0 747.0; 1208.0 766.0 863.0] | [1184.0 1359.0 1037.0; 1054.0 1032.0 872.0; 1152.0 916.0 1018.0] | [1214.0 1273.0 1059.0; 1105.0 1051.0 1101.0; 1121.0 979.0 1180.0] | [1215.0 1225.0 1086.0; 1203.0 1025.0 1241.0; 1108.0 1125.0 1190.0] | [1213.0 1202.0 1089.0; 1223.0 1127.0 1316.0; 1152.0 1188.0 1202.0] | [1225.0 1183.0 1134.0; 1234.0 1288.0 1289.0; 1104.0 1148.0 1285.0] | [1190.0 1206.0 1193.0; 1196.0 1335.0 1182.0; 1095.0 1156.0 1307.0] |
| **352** | [1505.0 1452.0 916.0; 1285.0 803.0 1408.0; 958.0 1101.0 896.0] | [1230.0 1355.0 1085.0; 1304.0 1095.0 1348.0; 1002.0 1088.0 991.0] | [1054.0 1062.0 1279.0; 1162.0 1205.0 1238.0; 870.0 1068.0 1298.0] | [1223.0 1032.0 1227.0; 1245.0 1299.0 1160.0; 1031.0 1175.0 1355.0] | [1414.0 1241.0 1091.0; 1406.0 1357.0 1180.0; 1204.0 1290.0 1186.0] | [1368.0 1350.0 1089.0; 1360.0 1268.0 1300.0; 1255.0 1274.0 1154.0] | [1275.0 1312.0 1080.0; 1270.0 1229.0 1351.0; 1300.0 1135.0 1214.0] | [1201.0 1301.0 1163.0; 1269.0 1217.0 1285.0; 1272.0 1061.0 1186.0] |
| **353** | [1290.0 889.0 1183.0; 1408.0 673.0 | [1281.0 1061.0 1176.0; 989.0 949.0 | [1287.0 1128.0 1065.0; 990.0 1215.0 [illegible]8.0; | [1300.0 1142.0 1207.0; 1143.0 1442.0 1052.0; | [1288.0 1155.0 1378.0; 1150.0 1348.0 1115.0; | [1341.0 1157.0 1321.0; 1116.0 1246.0 1212.0; | [1352.0 1210.0 1269.0; 1156.0 1308.0 1218.0; | [1311.0 1228.0 1240.0; 1208.0 1415.0 1202.0; |

| Row | V1 Array… | V2 Array… | V3 Array… | V4 Array… | V5 Array… | V6 Array… | V7 Array… | V8 Array… |
|---|---|---|---|---|---|---|---|---|
| | 1292.0 1147.0 992.0] | 1076.0 1165.0 941.0] | 1029.0 1136.0 880.0] | 1199.0 1148.0 1084.0] | 1284.0 1196.0 1207.0] | 1216.0 1207.0 1204.0] | 1199.0 1231.0 1167.0] | 1140.0 1257.0 1242.0] |
| **354** | [728.0 888.0 643.0; 162.0 586.0 1005.0; 519.0 955.0 572.0] | [493.0 729.0 754.0; 485.0 367.0 845.0; 333.0 526.0 582.0] | [458.0 560.0 663.0; 570.0 395.0 599.0; 389.0 457.0 618.0] | [522.0 431.0 546.0; 573.0 466.0 542.0; 348.0 432.0 503.0] | [542.0 304.0 498.0; 486.0 467.0 467.0; 321.0 368.0 404.0] | [374.0 399.0 459.0; 464.0 437.0 372.0; 328.0 332.0 315.0] | [337.0 465.0 380.0; 434.0 388.0 318.0; 294.0 308.0 262.0] | [432.0 405.0 404.0; 361.0 376.0 313.0; 261.0 272.0 278.0] |
| **355** | [462.0 546.0 801.0; 168.0 350.0 1261.0; 570.0 813.0 1019.0] | [539.0 362.0 800.0; 171.0 603.0 861.0; 476.0 669.0 966.0] | [576.0 235.0 748.0; 396.0 617.0 376.0; 459.0 586.0 693.0] | [514.0 210.0 621.0; 515.0 583.0 261.0; 404.0 435.0 548.0] | [442.0 241.0 511.0; 576.0 497.0 258.0; 334.0 404.0 568.0] | [396.0 314.0 468.0; 383.0 327.0 395.0; 310.0 366.0 586.0] | [322.0 316.0 453.0; 282.0 242.0 480.0; 389.0 335.0 465.0] | [153.0 252.0 391.0; 308.0 214.0 430.0; 411.0 366.0 350.0] |
| **356** | [1072.0 652.0 1195.0; 1118.0 1023.0 561.0; 1557.0 780.0 609.0] | [1049.0 724.0 962.0; 1128.0 1006.0 651.0; 1491.0 663.0 496.0] | [900.0 779.0 716.0; 1053.0 832.0 496.0; 1279.0 656.0 484.0] | [716.0 769.0 715.0; 913.0 718.0 351.0; 1419.0 706.0 328.0] | [739.0 747.0 771.0; 907.0 655.0 318.0; 1488.0 705.0 277.0] | [816.0 664.0 700.0; 995.0 553.0 263.0; 1386.0 616.0 299.0] | [818.0 609.0 527.0; 964.0 475.0 249.0; 1452.0 509.0 288.0] | [751.0 631.0 355.0; 846.0 446.0 302.0; 1550.0 453.0 268.0] |
| **357** | [1310.0 727.0 908.0; 759.0 824.0 758.0; 607.0 804.0 672.0] | [1364.0 780.0 947.0; 524.0 676.0 652.0; 402.0 763.0 400.0] | [1079.0 612.0 904.0; 539.0 450.0 429.0; 268.0 617.0 433.0] | [979.0 617.0 708.0; 532.0 460.0 400.0; 272.0 477.0 476.0] | [989.0 666.0 546.0; 585.0 511.0 398.0; 280.0 327.0 426.0] | [984.0 703.0 520.0; 612.0 494.0 388.0; 250.0 341.0 321.0] | [893.0 635.0 535.0; 480.0 447.0 456.0; 316.0 324.0 357.0] | [884.0 542.0 488.0; 425.0 338.0 433.0; 292.0 251.0 382.0] |
| **358** | [1980.0 1828.0 1224.0; 1003.0 859.0 1056.0; 1269.0 780.0 521.0] | [2001.0 1745.0 1178.0; 1065.0 635.0 929.0; 878.0 1013.0 346.0] | [2061.0 1746.0 1217.0; 971.0 572.0 739.0; 651.0 667.0 340.0] | [2204.0 1681.0 1172.0; 932.0 593.0 637.0; 558.0 463.0 429.0] | [2433.0 1543.0 1096.0; 834.0 625.0 583.0; 506.0 518.0 354.0] | [2513.0 1641.0 864.0; 829.0 615.0 561.0; 468.0 523.0 351.0] | [2549.0 1848.0 841.0; 919.0 610.0 516.0; 369.0 441.0 435.0] | [2610.0 1919.0 929.0; 955.0 592.0 478.0; 381.0 382.0 391.0] |
| **359** | [1672.0 1170.0 1006.0; 1785.0 1247.0 1122.0; [illegible] | [1656.0 1129.0 873.0; 1746.0 1188.0 966.0; [illegible] | [1620.0 1006.0 765.0; 1694.0 1006.0 996.0; [illegible] | [1629.0 971.0 732.0; 1767.0 992.0 908.0; 1830.0 | [1691.0 852.0 725.0; 1808.0 1138.0 672.0; 1775.0 | [1689.0 762.0 809.0; 1781.0 1174.0 583.0; 1680.0 | [1826.0 824.0 819.0; 1827.0 1074.0 556.0; 1715.0 | [1933.0 926.0 683.0; 1843.0 1025.0 535.0; 1769.0 |

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

| Row | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 |
|---|---|---|---|---|---|---|---|---|
| | Array… | Array… | Array… | Array… | Array… | Array… | Array… | Array… |
| | 1431.0 748.0] | 1426.0 632.0] | 1272.0 452.0] | 1115.0 491.0] | 1110.0 540.0] | 1186.0 564.0] | 1069.0 468.0] | 947.0 463.0] |
| **360** | [1206.0 494.0 969.0; 1037.0 587.0 628.0; 503.0 382.0 928.0] | [1140.0 818.0 659.0; 879.0 698.0 586.0; 439.0 525.0 693.0] | [899.0 840.0 692.0; 792.0 481.0 522.0; 345.0 588.0 454.0] | [784.0 739.0 743.0; 808.0 305.0 387.0; 339.0 509.0 247.0] | [769.0 779.0 612.0; 778.0 292.0 367.0; 370.0 278.0 234.0] | [720.0 813.0 508.0; 709.0 283.0 367.0; 412.0 287.0 163.0] | [585.0 809.0 479.0; 632.0 274.0 368.0; 373.0 306.0 84.0] | [606.0 792.0 384.0; 512.0 332.0 341.0; 303.0 247.0 111.0] |

In [50]:

```
# Let's unwind the spatial axes
X_df_static = Matrix(X_df)
cols = []
for i_var in 1:size(X_df_static, 2)
    var_unroll = cat(X_df_static[:,i_var]...; dims = 3)
    append!(cols, eachrow(reshape(var_unroll, (9, nrow(X_df)))))
end
X_df_static = DataFrame(cols, ["$n[$i][$j]" for n in names(X_df) for i in 1:
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

Out[50]: 360×927 DataFrame *827 columns and 335 rows omitted*

| Row | V1[1][1] | V1[1][2] | V1[1][3] | V1[2][1] | V1[2][2] | V1[2][3] | V1[3][1] | V1[3][2] |
|---|---|---|---|---|---|---|---|---|
| | Float64 | Float64 | Float64 | Float64 | Float64 | Float64 | Float64 | Float6 |
| **1** | 1186.0 | 1701.0 | 1823.0 | 1294.0 | 1705.0 | 1359.0 | 1291.0 | 1373. |
| **2** | 1153.0 | 784.0 | 1173.0 | 1201.0 | 1618.0 | 1071.0 | 1563.0 | 1508. |
| **3** | 880.0 | 672.0 | 540.0 | 804.0 | 865.0 | 757.0 | 727.0 | 1067. |
| **4** | 1454.0 | 512.0 | 1430.0 | 1082.0 | 985.0 | 1400.0 | 1291.0 | 936. |
| **5** | 1603.0 | 1003.0 | 1222.0 | 1091.0 | 1260.0 | 801.0 | 594.0 | 891. |
| **6** | 966.0 | 482.0 | 427.0 | 632.0 | 669.0 | 811.0 | 605.0 | 511. |
| **7** | 1219.0 | 1377.0 | 1187.0 | 1459.0 | 483.0 | 773.0 | 1247.0 | 1501. |
| **8** | 510.0 | 637.0 | 815.0 | 686.0 | 748.0 | 686.0 | 863.0 | 988. |
| **9** | 541.0 | 857.0 | 1317.0 | 814.0 | 1166.0 | 620.0 | 1198.0 | 812. |
| **10** | 856.0 | 463.0 | 631.0 | 302.0 | 460.0 | 762.0 | 568.0 | 870. |
| **11** | 589.0 | 1055.0 | 747.0 | 838.0 | 877.0 | 1016.0 | 602.0 | 1041. |
| **12** | 967.0 | 1147.0 | 907.0 | 1086.0 | 1269.0 | 1052.0 | 1269.0 | 1194. |
| **13** | 663.0 | 627.0 | 893.0 | 1111.0 | 642.0 | 474.0 | 382.0 | 816. |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| **349** | 2669.0 | 2567.0 | 2279.0 | 3073.0 | 2651.0 | 2681.0 | 2410.0 | 2144. |
| **350** | 3427.0 | 2280.0 | 2478.0 | 3340.0 | 2584.0 | 2516.0 | 2515.0 | 1827. |
| **351** | 1300.0 | 969.0 | 1005.0 | 1477.0 | 1080.0 | 591.0 | 1108.0 | 1134. |
| **352** | 1505.0 | 1285.0 | 958.0 | 1452.0 | 803.0 | 1101.0 | 916.0 | 1408. |
| **353** | 1290.0 | 1408.0 | 1292.0 | 889.0 | 673.0 | 1147.0 | 1183.0 | 778. |
| **354** | 728.0 | 162.0 | 519.0 | 888.0 | 586.0 | 955.0 | 643.0 | 1005. |
| **355** | 462.0 | 168.0 | 570.0 | 546.0 | 350.0 | 813.0 | 801.0 | 1261. |
| **356** | 1072.0 | 1118.0 | 1557.0 | 652.0 | 1023.0 | 780.0 | 1195.0 | 561. |
| **357** | 1310.0 | 759.0 | 607.0 | 727.0 | 824.0 | 804.0 | 908.0 | 758. |
| **358** | 1980.0 | 1003.0 | 1269.0 | 1828.0 | 859.0 | 780.0 | 1224.0 | 1056. |
| **359** | 1672.0 | 1785.0 | 1511.0 | 1170.0 | 1247.0 | 1431.0 | 1006.0 | 1122. |
| **360** | 1206.0 | 1037.0 | 503.0 | 494.0 | 587.0 | 382.0 | 969.0 | 628. |

In [51]:
```
using SoleData

X_multimodal = MultiModalDataset([X_df, X_df_static])
```

```
Out[51]: ● MultiModalDataset{DataFrame}
            └─ dimensionalities: (2, 0)
         - Modality 1 / 2
            └─ dimensionality: 2
         360×103 SubDataFrame
          Row │ V1                                V2
         V ⋯
              │ Array…                            Array…
         A ⋯
         ─────┼──────────────────────────────────────────────────────────────────
         ─────
            1 │ [1186.0 1294.0 1291.0; 1701.0 17…  [1069.0 1177.0 1369.0; 1708.0 14…
         [ ⋯
            2 │ [1153.0 1201.0 1563.0; 784.0 161…  [1024.0 1258.0 1480.0; 1024.0 15…
         [
            3 │ [880.0 804.0 727.0; 672.0 865.0 …  [726.0 486.0 771.0; 498.0 629.0 …
         [
            4 │ [1454.0 1082.0 1291.0; 512.0 985…  [1178.0 876.0 1229.0; 741.0 1014…
         [
            5 │ [1603.0 1091.0 594.0; 1003.0 126…  [1602.0 1092.0 689.0; 804.0 1240…
         [ ⋯
            6 │ [966.0 632.0 605.0; 482.0 669.0 …  [1171.0 630.0 451.0; 446.0 507.0…
         [
            7 │ [1219.0 1459.0 1247.0; 1377.0 48…  [1377.0 1259.0 1255.0; 1298.0 88…
         [
            8 │ [510.0 686.0 863.0; 637.0 748.0 …  [365.0 726.0 761.0; 810.0 639.0 …
         [
            9 │ [541.0 814.0 1198.0; 857.0 1166.…  [530.0 676.0 910.0; 697.0 848.0 …
         [ ⋯
           10 │ [856.0 302.0 568.0; 463.0 460.0 …  [698.0 291.0 598.0; 144.0 563.0 …
         [
           11 │ [589.0 838.0 602.0; 1055.0 877.0…  [671.0 617.0 508.0; 800.0 688.0 …
         [
          ⋮   │                 ⋮                                  ⋮
         ⋱
          351 │ [1300.0 1477.0 1108.0; 969.0 108…  [1289.0 1252.0 1139.0; 1172.0 10…
         [
          352 │ [1505.0 1452.0 916.0; 1285.0 803…  [1230.0 1355.0 1085.0; 1304.0 10…
         [ ⋯
          353 │ [1290.0 889.0 1183.0; 1408.0 673…  [1281.0 1061.0 1176.0; 989.0 949…
         [
          354 │ [728.0 888.0 643.0; 162.0 586.0 …  [493.0 729.0 754.0; 485.0 367.0 …
         [
          355 │ [462.0 546.0 801.0; 168.0 350.0 …  [539.0 362.0 800.0; 171.0 603.0 …
         [
          356 │ [1072.0 652.0 1195.0; 1118.0 102…  [1049.0 724.0 962.0; 1128.0 1006…
         [ ⋯
          357 │ [1310.0 727.0 908.0; 759.0 824.0…  [1364.0 780.0 947.0; 524.0 676.0…
         [
          358 │ [1980.0 1828.0 1224.0; 1003.0 85…  [2001.0 1745.0 1178.0; 1065.0 63…
         [
          359 │ [1672.0 1170.0 1006.0; 1785.0 12…  [1656.0 1129.0 873.0; 1746.0 118…
         [
          360 │ [1206.0 494.0 969.0; 1037.0 587.…  [1140.0 818.0 659.0; 879.0 698.0…
         [ ⋯
Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js      101 columns and 339 rows om
```

```
itted
- Modality 2 / 2
   └─ dimensionality: 0
360×927 SubDataFrame
 Row │ V1[1][1]  V1[1][2]  V1[1][3]  V1[2][1]  V1[2][2]  V1[2][3]  V1[3][1]
V …
     │ Float64   Float64   Float64   Float64   Float64   Float64   Float64
F …
─────┼────────────────────────────────────────────────────────────────────
─────
   1 │   1186.0    1701.0    1823.0    1294.0    1705.0    1359.0    1291.0
…
   2 │   1153.0     784.0    1173.0    1201.0    1618.0    1071.0    1563.0
   3 │    880.0     672.0     540.0     804.0     865.0     757.0     727.0
   4 │   1454.0     512.0    1430.0    1082.0     985.0    1400.0    1291.0
   5 │   1603.0    1003.0    1222.0    1091.0    1260.0     801.0     594.0
…
   6 │    966.0     482.0     427.0     632.0     669.0     811.0     605.0
   7 │   1219.0    1377.0    1187.0    1459.0     483.0     773.0    1247.0
   8 │    510.0     637.0     815.0     686.0     748.0     686.0     863.0
   9 │    541.0     857.0    1317.0     814.0    1166.0     620.0    1198.0
…
  10 │    856.0     463.0     631.0     302.0     460.0     762.0     568.0
  11 │    589.0    1055.0     747.0     838.0     877.0    1016.0     602.0
  ⋮  │    ⋮         ⋮         ⋮         ⋮         ⋮         ⋮
⋮         ⋱
 351 │   1300.0     969.0    1005.0    1477.0    1080.0     591.0    1108.0
 352 │   1505.0    1285.0     958.0    1452.0     803.0    1101.0     916.0
…
 353 │   1290.0    1408.0    1292.0     889.0     673.0    1147.0    1183.0
 354 │    728.0     162.0     519.0     888.0     586.0     955.0     643.0
 355 │    462.0     168.0     570.0     546.0     350.0     813.0     801.0
 356 │   1072.0    1118.0    1557.0     652.0    1023.0     780.0    1195.0
…
 357 │   1310.0     759.0     607.0     727.0     824.0     804.0     908.0
 358 │   1980.0    1003.0    1269.0    1828.0     859.0     780.0    1224.0
 359 │   1672.0    1785.0    1511.0    1170.0    1247.0    1431.0    1006.0
 360 │   1206.0    1037.0     503.0     494.0     587.0     382.0     969.0
…
                                                920 columns and 339 rows om
itted
```

In [52]:

```
using ModalDecisionTrees

model = ModalDecisionTree(; relations = :RCC8)
```

Out[52]:
```
ModalDecisionTree(
  max_depth = nothing,
  min_samples_leaf = 4,
  min_purity_increase = 0.002,
  max_purity_at_leaf = Inf,
  max_modal_depth = nothing,
  relations = :RCC8,
  features = nothing,
  conditions = nothing,
  featvaltype = Float64,
  initconditions = nothing,
  downsize = ModalDecisionTrees.MLJInterface.var"#downsize#43"(),
  print_progress = false,
  rng = Random._GLOBAL_RNG(),
  display_depth = nothing,
  min_samples_split = nothing,
  n_subfeatures = identity,
  post_prune = false,
  merge_purity_threshold = nothing,
  feature_importance = :split)
```

In [28]:
```
# Train in cross-validation!
e = @time evaluate!(machine(model, X_multimodal, y);
    resampling=StratifiedCV(rng = Random.Xoshiro(1), shuffle=true, nfolds =
    measures=[accuracy],
    verbosity=0,
    check_measure=false
)
```

```
[ Info: Precomputing logiset...
┌ Warning: ScalarOneStepMemoset: Found globalrel in relations in a single-wo
rld case.
└ @ SoleModels ~/.julia/packages/SoleModels/xvfwj/src/logisets/scalar/oneste
p-memoset.jl:304
694.598215 seconds (7.75 G allocations: 729.113 GiB, 11.43% gc time, 12.83%
compilation time: <1% of which was recompilation)
```

Out[28]:
```
PerformanceEvaluation object with these fields:
  model, measure, operation, measurement, per_fold,
  per_observation, fitted_params_per_fold,
  report_per_fold, train_test_rows, resampling, repeats
Extract:
┌────────────┬──────────────┬─────────────┬─────────┬──────────────────────
─────
│ measure    │ operation    │ measurement │ 1.96*SE │ per_fold
…
├────────────┼──────────────┼─────────────┼─────────┼──────────────────────
─────
│ Accuracy() │ predict_mode │ 0.775       │ 0.0537  │ [0.778, 0.833, 0.611,
0. …
└────────────┴──────────────┴─────────────┴─────────┴──────────────────────
─────
                                                                1 column om
itted
```

In [29]:
```julia
# Test accuracies per fold
e.per_fold
```

Out[29]:
```
1-element Vector{Vector{Float64}}:
 [0.7777777777777778, 0.8333333333333334, 0.6111111111111112, 0.91666666666
66666, 0.7777777777777778, 0.7777777777777778, 0.6944444444444444, 0.75, 0.
7777777777777778, 0.8333333333333334]
```

In [30]:
```julia
trees = map(((train_idxs, test_idxs), rep),)->begin
    predictions, tree_test = rep.sprinkle(slicedataset(X_multimodal, test_ic
    tree_test
end, zip(e.train_test_rows, e.report_per_fold))
```

```
Out[30]: 10-element Vector{DecisionTree{String, MultiFormula{Atom{ScalarCondition{In
t64, SoleModels.UnivariateValue, ScalarMetaCondition{SoleModels.UnivariateV
alue, typeof(>=)}}}}, SoleModels.ConstantModel{String}}}:
 ▣ {2}(V923 ≥ 1714)
├✔ {2}(V52 ≥ 1122)
│├✔ {2}(V248 ≥ 1882)
││├✔ Painted metal sheets
││└✘ {1}(⟨G⟩(min[V47] ≥ 2103))
││ ├✔ Gravel
││ └✘ {2}(V178 ≥ 1476)
││  ├✔ Self-Blocking Bricks
││  └✘ {1}(⟨G⟩(min[V2] < 1021))
││   ├✔ Self-Blocking Bricks
││   └✘ Gravel
│└✘ {1}(⟨G⟩(min[V68] < 563))
│ ├✔ Trees
│ └✘ {1}(⟨G⟩(min[V29] ≥ 1347))
│  ├✔ Bare Soil
│  └✘ {2}(V662 ≥ 1915)
│   ├✔ Meadows
│   └✘ {2}(V63 ≥ 579)
│    ├✔ Meadows
│    └✘ Bare Soil
└✘ {1}(⟨G⟩(min[V6] < 567))
 ├✔ Shadows
 └✘ {2}(V593 ≥ 1343)
  ├✔ {2}(V917 ≥ 1490)
  │├✔ Gravel
  │└✘ {1}(⟨G⟩(min[V3] < 754))
  │ ├✔ Asphalt
  │ └✘ Bitumen
  └✘ Asphalt

 ▣ {2}(V923 ≥ 1714)
├✔ {2}(V44 ≥ 1109)
│├✔ {1}(⟨G⟩(min[V38] ≥ 2424))
││├✔ Painted metal sheets
││└✘ {1}(⟨G⟩(min[V67] ≥ 2152))
││ ├✔ Gravel
││ └✘ {2}(V743 ≥ 1919)
││  ├✔ Self-Blocking Bricks
││  └✘ Gravel
│└✘ {1}(⟨G⟩(min[V44] < 543))
│ ├✔ {1}(⟨G⟩((min[V44] < 543) ∧ ⟨N̅T̅P̅P̅⟩(min[V28] ≥ 620)))
│ │├✔ Meadows
│ │└✘ Trees
│ └✘ {1}(⟨G⟩(min[V29] ≥ 1347))
│  ├✔ Bare Soil
│  └✘ {2}(V662 ≥ 1908)
│   ├✔ Meadows
│   └✘ Bare Soil
└✘ {1}(⟨G⟩(min[V6] < 567))
 ├✔ Shadows
 └✘ {2}(V593 ≥ 1343)
  ├✔ {2}(V667 ≥ 1686)
```
Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

```
  │└✘ {2}(V201 ≥ 1266)
  │ ├✔ Bitumen
  │ └✘ Asphalt
  └✘ Asphalt

 ▣ {2}(V923 ≥ 1714)
├✔ {2}(V52 ≥ 1114)
│├✔ {1}(⟨G⟩(min[V38] ≥ 2267))
││├✔ {1}(⟨G⟩((min[V38] ≥ 2267) ∧ (min[V1] ≥ 2642)))
│││├✔ Painted metal sheets
│││└✘ Bare Soil
││└✘ {2}(V167 ≥ 1475)
││ ├✔ {1}(⟨G⟩(min[V4] < 1185))
││ │├✔ Self-Blocking Bricks
││ │└✘ Gravel
││ └✘ Gravel
│└✘ {1}(⟨G⟩(min[V68] < 563))
│ ├✔ Trees
│ └✘ {1}(⟨G⟩(min[V29] ≥ 1347))
│  ├✔ Bare Soil
│  └✘ {2}(V877 ≥ 2483)
│   ├✔ Meadows
│   └✘ Bare Soil
└✘ {1}(⟨G⟩(min[V5] < 516))
 ├✔ Shadows
 └✘ {2}(V593 ≥ 1343)
  ├✔ {2}(V917 ≥ 1490)
  │├✔ {2}(V106 ≥ 1346)
  ││├✔ Bitumen
  ││└✘ Gravel
  │└✘ Bitumen
  └✘ Asphalt

 ▣ {2}(V923 ≥ 1714)
├✔ {2}(V52 ≥ 1114)
│├✔ {2}(V248 ≥ 1882)
││├✔ {1}(⟨G⟩(min[V1] ≥ 2642))
│││├✔ Painted metal sheets
│││└✘ Bare Soil
││└✘ {1}(⟨G⟩(min[V48] ≥ 2114))
││ ├✔ Gravel
││ └✘ {2}(V742 ≥ 1915)
││  ├✔ Self-Blocking Bricks
││  └✘ {2}(V132 ≥ 1394)
││   ├✔ Self-Blocking Bricks
││   └✘ Gravel
│└✘ {2}(V540 ≥ 428)
│ ├✔ Meadows
│ └✘ Trees
└✘ {1}(⟨G⟩(min[V6] < 567))
 ├✔ Shadows
 └✘ {2}(V593 ≥ 1343)
  ├✔ {2}(V876 ≥ 1600)
  │├✔ Gravel
  │└✘ {2}(V155 ≥ 1168)
  │ ├✔ Bitumen
```

```
  │ └✘ Asphalt
  └✘ Asphalt

 ▣ {2}(V923 ≥ 1715)
├✔ {2}(V34 ≥ 1009)
│├✔ {2}(V248 ≥ 1882)
││├✔ {1}(⟨G⟩(min[V1] ≥ 2642))
│││├✔ Painted metal sheets
│││└✘ Bare Soil
││└✘ {1}(⟨G⟩(min[V48] ≥ 2114))
││ ├✔ Gravel
││ └✘ {2}(V197 ≥ 1459)
││  ├✔ Self-Blocking Bricks
││  └✘ Gravel
│└✘ {2}(V540 ≥ 428)
│ ├✔ {1}(⟨G⟩(min[V29] ≥ 1347))
│ │├✔ Bare Soil
│ │└✘ {2}(V662 ≥ 1908)
│ │ ├✔ Meadows
│ │ └✘ {1}(⟨G⟩(min[V37] ≥ 891))
│ │  ├✔ Bare Soil
│ │  └✘ Meadows
│ └✘ {1}(⟨G⟩(min[V72] < 1534))
│  ├✔ Trees
│  └✘ Meadows
└✘ {1}(⟨G⟩(min[V6] < 597))
 ├✔ Shadows
 └✘ {2}(V593 ≥ 1343)
  ├✔ {2}(V876 ≥ 1600)
  │├✔ Gravel
  │└✘ {1}(⟨G⟩(min[V18] ≥ 1318))
  │ ├✔ Bitumen
  │ └✘ Asphalt
  └✘ Asphalt

 ▣ {2}(V860 ≥ 1700)
├✔ {2}(V52 ≥ 1114)
│├✔ {1}(⟨G⟩(min[V38] ≥ 2424))
││├✔ {1}(⟨G⟩((min[V38] ≥ 2424) ∧ (min[V1] ≥ 2769)))
│││├✔ Painted metal sheets
│││└✘ Bare Soil
││└✘ {1}(⟨G⟩(min[V48] ≥ 2114))
││ ├✔ Gravel
││ └✘ {2}(V743 ≥ 1919)
││  ├✔ Self-Blocking Bricks
││  └✘ Gravel
│└✘ {2}(V603 ≥ 606)
│ ├✔ Meadows
│ └✘ Trees
└✘ {1}(⟨G⟩(min[V6] < 567))
 ├✔ Shadows
 └✘ {2}(V593 ≥ 1343)
  ├✔ {2}(V142 ≥ 1281)
  │├✔ Bitumen
  │└✘ {2}(V917 ≥ 1401)
  │ ├✔ Gravel
```

```
  │ └✘ Bitumen
  └✘ Asphalt

 ▣ {2}(V923 ≥ 1714)
├✔ {2}(V52 ≥ 1114)
│├✔ {2}(V248 ≥ 1882)
││├✔ {1}(⟨G⟩(min[V1] ≥ 2642))
│││├✔ Painted metal sheets
│││└✘ Bare Soil
││└✘ Self-Blocking Bricks
│└✘ {2}(V540 ≥ 428)
│ ├✔ {1}(⟨G⟩(min[V29] ≥ 1347))
│ │├✔ Bare Soil
│ │└✘ {1}(⟨G⟩(min[V85] < 1923))
│ │ ├✔ Bare Soil
│ │ └✘ {2}(V6 ≥ 969)
│ │  ├✔ Bare Soil
│ │  └✘ Meadows
│ └✘ {1}(⟨G⟩(min[V72] < 1534))
│  ├✔ Trees
│  └✘ Meadows
└✘ {1}(⟨G⟩(min[V6] < 567))
 ├✔ Shadows
 └✘ {2}(V593 ≥ 1343)
  ├✔ {2}(V415 ≥ 1666)
  │├✔ Gravel
  │└✘ Bitumen
  └✘ Asphalt

 ▣ {2}(V923 ≥ 1714)
├✔ {2}(V52 ≥ 1114)
│├✔ {2}(V248 ≥ 1882)
││├✔ Painted metal sheets
││└✘ Self-Blocking Bricks
│└✘ {2}(V477 ≥ 482)
│ ├✔ {1}(⟨G⟩(min[V29] ≥ 1348))
│ │├✔ Bare Soil
│ │└✘ {2}(V762 ≥ 2450)
│ │ ├✔ Meadows
│ │ └✘ Bare Soil
│ └✘ Trees
└✘ {1}(⟨G⟩(min[V4] < 496))
 ├✔ Shadows
 └✘ {2}(V575 ≥ 1369)
  ├✔ {2}(V876 ≥ 1600)
  │├✔ Gravel
  │└✘ {2}(V155 ≥ 1168)
  │ ├✔ Bitumen
  │ └✘ Asphalt
  └✘ Asphalt

 ▣ {2}(V923 ≥ 1714)
├✔ {2}(V52 ≥ 1114)
│├✔ {2}(V239 ≥ 1857)
││├✔ {1}(⟨G⟩(min[V1] ≥ 2642))
│││├✔ Painted metal sheets
```

```
│││└✘ Bare Soil
││└✘ {2}(V170 ≥ 1534)
││ ├✔ Self-Blocking Bricks
││ └✘ {2}(V14 ≥ 1205)
││   ├✔ Gravel
││   └✘ {2}(V528 ≥ 1925)
││     ├✔ Gravel
││     └✘ Self-Blocking Bricks
│└✘ {2}(V540 ≥ 428)
│ ├✔ {1}(⟨G⟩(min[V28] ≥ 1302))
│ │├✔ Bare Soil
│ │└✘ {1}(⟨G⟩(min[V102] < 2299))
│ │ ├✔ Meadows
│ │ └✘ Meadows
│ └✘ Trees
└✘ {1}(⟨G⟩(min[V6] < 567))
 ├✔ Shadows
 └✘ {2}(V593 ≥ 1343)
  ├✔ Bitumen
  └✘ Asphalt

▣ {2}(V923 ≥ 1649)
├✔ {2}(V52 ≥ 1114)
│├✔ {2}(V248 ≥ 1882)
││├✔ {1}(⟨G⟩(min[V1] ≥ 2642))
│││├✔ Painted metal sheets
│││└✘ Bare Soil
││└✘ {2}(V141 ≥ 1464)
││ ├✔ Self-Blocking Bricks
││ └✘ Gravel
│└✘ {2}(V540 ≥ 428)
│ ├✔ {1}(⟨G⟩(min[V29] ≥ 1347))
│ │├✔ Bare Soil
│ │└✘ {1}(⟨G⟩(min[V102] < 2299))
│ │ ├✔ Bare Soil
│ │ └✘ Meadows
│ └✘ Trees
└✘ {1}(⟨G⟩(min[V6] < 567))
 ├✔ Shadows
 └✘ {2}(V566 ≥ 1385)
  ├✔ {2}(V845 ≥ 1661)
  │├✔ Gravel
  │└✘ {2}(V175 ≥ 1261)
  │ ├✔ Bitumen
  │ └✘ Asphalt
  └✘ Asphalt
```

In [31]:

```
rules = vcat(listrules.(trees)...)
```

Out[31]:
```
134-element Vector{Rule{String, A, SoleModels.ConstantModel{String}} where
A<:Formula}:
 ▣ {2}(V248 ≥ 1882)  ➡  Painted metal sheets

 ▣ {1}((⟨G⟩(min[V47] ≥ 2103)) ∧ {2}(V52 ≥ 1122) ∧ (V248 < 1882)  ➡  Gravel

 ▣ {1}([G](min[V47] < 2103)) ∧ {2}(V178 ≥ 1476) ∧ (V248 < 1882)  ➡  Self-Bl
ocking Bricks

 ▣ {1}⟨G⟩(min[V2] < 1021) ∧ [G](min[V47] < 2103) ∧ {2}(V52 ≥ 1122) ∧ (V248
< 1882) ∧ (V178 < 1476)  ➡  Self-Blocking Bricks

 ▣ {1}[G](min[V47] < 2103) ∧ [G](min[V2] ≥ 1021) ∧ {2}(V52 ≥ 1122) ∧ (V248
< 1882) ∧ (V178 < 1476)  ➡  Gravel

 ▣ {1}((⟨G⟩(min[V68] < 563)) ∧ {2}(V923 ≥ 1714) ∧ (V52 < 1122)  ➡  Trees

 ▣ {1}⟨G⟩(min[V29] ≥ 1347) ∧ [G](min[V68] ≥ 563) ∧ {2}(V923 ≥ 1714) ∧ (V52
< 1122)  ➡  Bare Soil

 ▣ {1}[G](min[V68] ≥ 563) ∧ [G](min[V29] < 1347) ∧ {2}(V662 ≥ 1915) ∧ (V52
< 1122)  ➡  Meadows

 ▣ {1}[G](min[V68] ≥ 563) ∧ [G](min[V29] < 1347) ∧ {2}(V63 ≥ 579) ∧ (V52 <
1122) ∧ (V662 < 1915)  ➡  Meadows

 ▣ {1}[G](min[V68] ≥ 563) ∧ [G](min[V29] < 1347) ∧ {2}(V923 ≥ 1714) ∧ (V52
< 1122) ∧ (V662 < 1915) ∧ (V63 < 579)  ➡  Bare Soil

 ▣ {1}((⟨G⟩(min[V6] < 567)) ∧ {2}(V923 < 1714)  ➡  Shadows

 ▣ {1}([G](min[V6] ≥ 567)) ∧ {2}(V917 ≥ 1490) ∧ (V923 < 1714)  ➡  Gravel

 ▣ {1}⟨G⟩(min[V3] < 754) ∧ [G](min[V6] ≥ 567) ∧ {2}(V593 ≥ 1343) ∧ (V923 <
1714) ∧ (V917 < 1490)  ➡  Asphalt

  ⋮
 ▣ {1}([G](min[V1] < 2642)) ∧ {2}(V248 ≥ 1882)  ➡  Bare Soil

 ▣ {2}(V141 ≥ 1464) ∧ (V248 < 1882)  ➡  Self-Blocking Bricks

 ▣ {2}(V52 ≥ 1114) ∧ (V248 < 1882) ∧ (V141 < 1464)  ➡  Gravel

 ▣ {1}((⟨G⟩(min[V29] ≥ 1347)) ∧ {2}(V540 ≥ 428) ∧ (V52 < 1114)  ➡  Bare Soil

 ▣ {1}⟨G⟩(min[V102] < 2299) ∧ [G](min[V29] < 1347) ∧ {2}(V540 ≥ 428) ∧ (V52
< 1114)  ➡  Bare Soil

 ▣ {1}[G](min[V29] < 1347) ∧ [G](min[V102] ≥ 2299) ∧ {2}(V540 ≥ 428) ∧ (V52
< 1114)  ➡  Meadows

 ▣ {2}(V923 ≥ 1649) ∧ (V52 < 1114) ∧ (V540 < 428)  ➡  Trees

 ▣ {1}((⟨G⟩(min[V6] < 567)) ∧ {2}(V923 < 1649)  ➡  Shadows

) ∧ {2}(V845 ≥ 1661) ∧ (V923 < 1649)  ➡  Gravel
```

▣ {1}([G](min[V6] ≥ 567)) ∧ {2}(V175 ≥ 1261) ∧ (V923 < 1649) ∧ (V845 < 1661) ➡ Bitumen

▣ {1}([G](min[V6] ≥ 567)) ∧ {2}(V566 ≥ 1385) ∧ (V923 < 1649) ∧ (V845 < 1661) ∧ (V175 < 1261) ➡ Asphalt

▣ {1}([G](min[V6] ≥ 567)) ∧ {2}(V923 < 1649) ∧ (V566 < 1385) ➡ Asphalt

In [32]:
```
# Every symbolic model (including rules) can have has additional information
println(rules[1])

ruleinfo = SoleModels.info(rules[1])
println(keys(ruleinfo))
```

```
▣ {2}(V248 ≥ 1882) ➡ Painted metal sheets

(:supporting_labels, :supporting_predictions, :shortform)
```

In [33]:
```
ruleinfo[:supporting_predictions] |> length
```

Out[33]:
```
4
```

In [34]:
```
sort(readmetrics.(rules), by=x->x[:coverage], rev = true)
```

Out[34]:
```
134-element Vector{NamedTuple{(:ninstances, :confidence, :coverage), Tuple{Int64, Float64, Float64}}}:
 (ninstances = 7, confidence = 0.57, coverage = 0.19)
 (ninstances = 6, confidence = 0.67, coverage = 0.17)
 (ninstances = 6, confidence = 0.67, coverage = 0.17)
 (ninstances = 6, confidence = 0.67, coverage = 0.17)
 (ninstances = 6, confidence = 0.67, coverage = 0.17)
 (ninstances = 5, confidence = 0.4, coverage = 0.14)
 (ninstances = 5, confidence = 0.4, coverage = 0.14)
 (ninstances = 5, confidence = 0.8, coverage = 0.14)
 (ninstances = 5, confidence = 0.6, coverage = 0.14)
 (ninstances = 5, confidence = 0.8, coverage = 0.14)
 (ninstances = 5, confidence = 0.8, coverage = 0.14)
 (ninstances = 5, confidence = 0.6, coverage = 0.14)
 (ninstances = 5, confidence = 0.6, coverage = 0.14)
 ⋮
 (ninstances = 1, confidence = 0.0, coverage = 0.03)
 (ninstances = 1, confidence = 0.0, coverage = 0.03)
 (ninstances = 1, confidence = 1.0, coverage = 0.03)
 (ninstances = 1, confidence = 0.0, coverage = 0.03)
 (ninstances = 1, confidence = 1.0, coverage = 0.03)
 (ninstances = 1, confidence = 1.0, coverage = 0.03)
 (ninstances = 1, confidence = 0.0, coverage = 0.03)
 (ninstances = 1, confidence = 1.0, coverage = 0.03)
 (ninstances = 1, confidence = 0.0, coverage = 0.03)
 (ninstances = 1, confidence = 1.0, coverage = 0.03)
 (ninstances = 1, confidence = 0.0, coverage = 0.03)
 (ninstances = 1, confidence = 1.0, coverage = 0.03)
```

In [53]:
```
# goodrules = filter(r->readmetrics(r)[:ninstances] > 1, rules)
goodrules = sort(rules, by=r->readmetrics(r)[:coverage], rev = true)
```

```
printmodel.(goodrules; show_metrics = true, threshold_digits = 4);
```

▣ {2}(V52 ≥ 1114.0) ∧ (V248 < 1882.0) ➡ Self-Blocking Bricks : (ninstances = 7, confidence = 0.57, coverage = 0.19)

▣ {1}⟨G⟩(min[V44] < 543.0) ∧ [G]((min[V44] < 543.0) → [$\overline{\text{NTPP}}$](min[V28] < 620.0)) ∧ {2}(V923 ≥ 1714.0) ∧ (V44 < 1109.0) ➡ Trees : (ninstances = 6, confidence = 0.67, coverage = 0.17)

▣ {2}(V540 ≥ 428.0) ∧ (V52 < 1114.0) ➡ Meadows : (ninstances = 6, confidence = 0.67, coverage = 0.17)

▣ {2}(V52 ≥ 1114.0) ∧ (V248 < 1882.0) ➡ Self-Blocking Bricks : (ninstances = 6, confidence = 0.67, coverage = 0.17)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V593 ≥ 1343.0) ∧ (V923 < 1714.0) ➡ Bitumen : (ninstances = 6, confidence = 0.67, coverage = 0.17)

▣ {1}[G](min[V68] ≥ 563.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V923 ≥ 1714.0) ∧ (V52 < 1114.0) ∧ (V877 < 2483.0) ➡ Bare Soil : (ninstances = 5, confidence = 0.4, coverage = 0.14)

▣ {1}([G](min[V48] < 2114.0)) ∧ {2}(V197 ≥ 1459.0) ∧ (V248 < 1882.0) ➡ Self-Blocking Bricks : (ninstances = 5, confidence = 0.4, coverage = 0.14)

▣ {1}(⟨G⟩(min[V6] < 597.0)) ∧ {2}(V923 < 1715.0) ➡ Shadows : (ninstances = 5, confidence = 0.8, coverage = 0.14)

▣ {1}[G](min[V38] < 2424.0) ∧ [G](min[V48] < 2114.0) ∧ {2}(V52 ≥ 1114.0) ∧ (V743 < 1919.0) ➡ Gravel : (ninstances = 5, confidence = 0.6, coverage = 0.14)

▣ {2}(V603 ≥ 606.0) ∧ (V52 < 1114.0) ➡ Meadows : (ninstances = 5, confidence = 0.8, coverage = 0.14)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V593 ≥ 1343.0) ∧ (V923 < 1714.0) ∧ (V415 < 1666.0) ➡ Bitumen : (ninstances = 5, confidence = 0.8, coverage = 0.14)

▣ {1}([G](min[V29] < 1348.0)) ∧ {2}(V762 ≥ 2450.0) ∧ (V52 < 1114.0) ➡ Meadows : (ninstances = 5, confidence = 0.6, coverage = 0.14)

▣ {1}([G](min[V4] ≥ 496.0)) ∧ {2}(V923 < 1714.0) ∧ (V575 < 1369.0) ➡ Asphalt : (ninstances = 5, confidence = 0.6, coverage = 0.14)

▣ {2}(V248 ≥ 1882.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}[G](min[V68] ≥ 563.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V662 ≥ 1915.0) ∧ (V52 < 1122.0) ➡ Meadows : (ninstances = 4, confidence = 0.5, coverage = 0.11)

▣ {1}(⟨G⟩(min[V6] < 567.0)) ∧ {2}(V923 < 1714.0) ➡ Shadows : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}(⟨G⟩(min[V38] ≥ 2424.0)) ∧ {2}(V44 ≥ 1109.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}(⟨G⟩(min[V6] < 567.0)) ∧ {2}(V923 < 1714.0) ➡ Shadows : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V923 < 1714.0) ∧ (V593 < 1343.0) ➡ Asphalt : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}(⟨G⟩((min[V38] ≥ 2267.0) ∧ (min[V1] ≥ 2642.0))) ∧ {2}(V52 ≥ 1114.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}([G](min[V5] ≥ 516.0)) ∧ {2}(V106 ≥ 1346.0) ∧ (V923 < 1714.0) ➡ Bitumen : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}([G](min[V5] ≥ 516.0)) ∧ {2}(V923 < 1714.0) ∧ (V593 < 1343.0) ➡ Asphalt : (ninstances = 4, confidence = 0.75, coverage = 0.11)

▣ {1}(⟨G⟩(min[V1] ≥ 2642.0)) ∧ {2}(V248 ≥ 1882.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {2}(V923 ≥ 1714.0) ∧ (V52 < 1114.0) ∧ (V540 < 428.0) ➡ Trees : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}(⟨G⟩(min[V6] < 567.0)) ∧ {2}(V923 < 1714.0) ➡ Shadows : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V155 ≥ 1168.0) ∧ (V923 < 1714.0) ∧ (V876 < 1690.0) ➡ Bitumen : (ninstances = 4, confidence = 1.0, coverage = 0.11)

▣ {1}(⟨G⟩(min[V1] ≥ 2642.0)) ∧ {2}(V248 ≥ 1882.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}(⟨G⟩(min[V72] < 1534.0)) ∧ {2}(V923 ≥ 1715.0) ∧ (V34 < 1009.0) ∧ (V540 < 428.0) ➡ Trees : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}([G](min[V6] ≥ 597.0)) ∧ {2}(V923 < 1715.0) ∧ (V593 < 1343.0) ➡ Asphalt : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}(⟨G⟩(min[V6] < 567.0)) ∧ {2}(V860 < 1700.0) ➡ Shadows : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V860 < 1700.0) ∧ (V593 < 1343.0) ➡ Asphalt : (ninstances = 4, confidence = 0.75, coverage = 0.11)
▣ {1}(⟨G⟩(min[V1] ≥ 2642.0)) ∧ {2}(V248 ≥ 1882.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}(⟨G⟩(min[V72] < 1534.0)) ∧ {2}(V923 ≥ 1714.0) ∧ (V52 < 1114.0) ∧ (V540 < 428.0) ➡ Trees : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}(⟨G⟩(min[V6] < 567.0)) ∧ {2}(V923 < 1714.0) ➡ Shadows : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {2}(V248 ≥ 1882.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}([G](min[V4] ≥ 496.0)) ∧ {2}(V155 ≥ 1168.0) ∧ (V923 < 1714.0) ∧ (V876 < 1600.0) ➡ Bitumen : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}(⟨G⟩(min[V1] ≥ 2642.0)) ∧ {2}(V239 ≥ 1857.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {2}(V923 ≥ 1714.0) ∧ (V52 < 1114.0) ∧ (V540 < 428.0) ➡ Trees : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}(⟨G⟩(min[V6] < 567.0)) ∧ {2}(V923 < 1714.0) ➡ Shadows : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}(⟨G⟩(min[V1] ≥ 2642.0)) ∧ {2}(V248 ≥ 1882.0) ➡ Painted metal sheets : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {2}(V141 ≥ 1464.0) ∧ (V248 < 1882.0) ➡ Self-Blocking Bricks : (ninstances = 4, confidence = 0.75, coverage = 0.11)
▣ {1}⟨G⟩(min[V102] < 2299.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V540 ≥ 428.0) ∧ (V52 < 1114.0) ➡ Bare Soil : (ninstances = 4, confidence = 0.25, coverage = 0.11)
▣ {2}(V923 ≥ 1649.0) ∧ (V52 < 1114.0) ∧ (V540 < 428.0) ➡ Trees : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}(⟨G⟩(min[V6] < 567.0)) ∧ {2}(V923 < 1649.0) ➡ Shadows : (ninstances = 4, confidence = 1.0, coverage = 0.11)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V923 < 1649.0) ∧ (V566 < 1385.0) ➡ Asphalt : (ninstances = 4, confidence = 0.75, coverage = 0.11)
▣ {1}(⟨G⟩(min[V68] < 563.0)) ∧ {2}(V923 ≥ 1714.0) ∧ (V52 < 1122.0) ➡ Trees : (ninstances = 3, confidence = 1.0, coverage = 0.08)
▣ {1}⟨G⟩(min[V29] ≥ 1347.0) ∧ [G](min[V68] ≥ 563.0) ∧ {2}(V923 ≥ 1714.0) ∧ (V52 < 1122.0) ➡ Bare Soil : (ninstances = 3, confidence = 0.67, coverage = 0.08)
▣ {1}[G](min[V6] ≥ 567.0) ∧ [G](min[V3] ≥ 754.0) ∧ {2}(V593 ≥ 1343.0) ∧ (V923 < 1714.0) ∧ (V917 < 1490.0) ➡ Bitumen : (ninstances = 3, confidence = 0.67, coverage = 0.08)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V923 < 1714.0) ∧ (V593 < 1343.0) ➡ Asphalt : (ninstances = 3, confidence = 1.0, coverage = 0.08)
▣ {1}⟨G⟩(min[V29] ≥ 1347.0) ∧ [G](min[V44] ≥ 543.0) ∧ {2}(V923 ≥ 1714.0) ∧ (V44 < 1109.0) ➡ Bare Soil : (ninstances = 3, confidence = 0.67, coverage = 0.08)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V201 ≥ 1266.0) ∧ (V923 < 1714.0) ∧ (V667 < 1686.0) ➡ Bitumen : (ninstances = 3, confidence = 1.0, coverage = 0.08)
▣ {1}[G](min[V68] ≥ 563.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V877 ≥ 2483.0) ∧

[illegible] : (ninstances = 3, confidence = 0.33, coverage =

0.08)
▣ {1}(⟨G⟩(min[V5] < 516.0)) ∧ {2}(V923 < 1714.0) ➡ Shadows : (ninstances =
3, confidence = 1.0, coverage = 0.08)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V923 < 1714.0) ∧ (V593 < 1343.0) ➡ Aspha
lt : (ninstances = 3, confidence = 1.0, coverage = 0.08)
▣ {1}⟨G⟩(min[V18] ≥ 1318.0) ∧ [G](min[V6] ≥ 597.0) ∧ {2}(V593 ≥ 1343.0) ∧ (V
923 < 1715.0) ∧ (V876 < 1600.0) ➡ Bitumen : (ninstances = 3, confidence =
1.0, coverage = 0.08)
▣ {1}(⟨G⟩((min[V38] ≥ 2424.0) ∧ (min[V1] ≥ 2769.0))) ∧ {2}(V52 ≥ 1114.0) ➡
Painted metal sheets : (ninstances = 3, confidence = 1.0, coverage = 0.08)
▣ {1}⟨G⟩(min[V38] ≥ 2424.0) ∧ [G]((min[V38] ≥ 2424.0) → (min[V1] < 2769.0))
∧ {2}(V52 ≥ 1114.0) ➡ Bare Soil : (ninstances = 3, confidence = 1.0, cover
age = 0.08)
▣ {2}(V860 ≥ 1700.0) ∧ (V52 < 1114.0) ∧ (V603 < 606.0) ➡ Trees : (ninstanc
es = 3, confidence = 1.0, coverage = 0.08)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V142 ≥ 1281.0) ∧ (V860 < 1700.0) ➡ Bitum
en : (ninstances = 3, confidence = 0.67, coverage = 0.08)
▣ {1}[G](min[V29] < 1347.0) ∧ [G](min[V85] ≥ 1923.0) ∧ {2}(V540 ≥ 428.0) ∧
(V52 < 1114.0) ∧ (V6 < 969.0) ➡ Meadows : (ninstances = 3, confidence = 0.
33, coverage = 0.08)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V923 < 1714.0) ∧ (V593 < 1343.0) ➡ Aspha
lt : (ninstances = 3, confidence = 1.0, coverage = 0.08)
▣ {2}(V923 ≥ 1714.0) ∧ (V52 < 1114.0) ∧ (V477 < 482.0) ➡ Trees : (ninstanc
es = 3, confidence = 1.0, coverage = 0.08)
▣ {1}(⟨G⟩(min[V4] < 496.0)) ∧ {2}(V923 < 1714.0) ➡ Shadows : (ninstances =
3, confidence = 1.0, coverage = 0.08)
▣ {2}(V170 ≥ 1534.0) ∧ (V239 < 1857.0) ➡ Self-Blocking Bricks : (ninstance
s = 3, confidence = 0.33, coverage = 0.08)
▣ {1}(⟨G⟩(min[V28] ≥ 1302.0)) ∧ {2}(V540 ≥ 428.0) ∧ (V52 < 1114.0) ➡ Bare
Soil : (ninstances = 3, confidence = 0.67, coverage = 0.08)
▣ {1}[G](min[V28] < 1302.0) ∧ [G](min[V102] ≥ 2299.0) ∧ {2}(V540 ≥ 428.0) ∧
(V52 < 1114.0) ➡ Meadows : (ninstances = 3, confidence = 1.0, coverage =
0.08)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V175 ≥ 1261.0) ∧ (V923 < 1649.0) ∧ (V845 <
1661.0) ➡ Bitumen : (ninstances = 3, confidence = 1.0, coverage = 0.08)
▣ {1}(⟨G⟩(min[V47] ≥ 2103.0)) ∧ {2}(V52 ≥ 1122.0) ∧ (V248 < 1882.0) ➡ Grav
el : (ninstances = 2, confidence = 0.5, coverage = 0.06)
▣ {1}([G](min[V47] < 2103.0)) ∧ {2}(V178 ≥ 1476.0) ∧ (V248 < 1882.0) ➡ Sel
f-Blocking Bricks : (ninstances = 2, confidence = 1.0, coverage = 0.06)
▣ {1}⟨G⟩(min[V2] < 1021.0) ∧ [G](min[V47] < 2103.0) ∧ {2}(V52 ≥ 1122.0) ∧ (V
248 < 1882.0) ∧ (V178 < 1476.0) ➡ Self-Blocking Bricks : (ninstances = 2,
confidence = 0.5, coverage = 0.06)
▣ {1}[G](min[V68] ≥ 563.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V923 ≥ 1714.0) ∧
(V52 < 1122.0) ∧ (V662 < 1915.0) ∧ (V63 < 579.0) ➡ Bare Soil : (ninstances
= 2, confidence = 0.0, coverage = 0.06)
▣ {1}⟨G⟩(min[V67] ≥ 2152.0) ∧ [G](min[V38] < 2424.0) ∧ {2}(V44 ≥ 1109.0) ➡
Gravel : (ninstances = 2, confidence = 0.5, coverage = 0.06)
▣ {1}[G](min[V38] < 2424.0) ∧ [G](min[V67] < 2152.0) ∧ {2}(V743 ≥ 1919.0) ➡
Self-Blocking Bricks : (ninstances = 2, confidence = 1.0, coverage = 0.06)
▣ {1}[G](min[V38] < 2424.0) ∧ [G](min[V67] < 2152.0) ∧ {2}(V44 ≥ 1109.0) ∧
(V743 < 1919.0) ➡ Gravel : (ninstances = 2, confidence = 0.5, coverage =
0.06)
▣ {1}(⟨G⟩((min[V44] < 543.0) ∧ ⟨$\overline{\text{NTPP}}$⟩(min[V28] ≥ 620.0))) ∧ {2}(V923 ≥ 1714.
0) ∧ (V44 < 1109.0) ➡ Meadows : (ninstances = 2, confidence = 1.0, coverag
e = 0.06)
▣ {1}⟨G⟩(min[V4] [illegible] 1185.0) ∧ [G](min[V38] < 2267.0) ∧ {2}(V167 ≥ 1475.0) ➡

Self-Blocking Bricks : (ninstances = 2, confidence = 0.5, coverage = 0.06)

▣ {1}([G](min[V38] < 2267.0)) ∧ {2}(V52 ≥ 1114.0) ∧ (V167 < 1475.0) ➡ Gravel : (ninstances = 2, confidence = 0.0, coverage = 0.06)

▣ {1}(⟨G⟩(min[V68] < 563.0)) ∧ {2}(V923 ≥ 1714.0) ∧ (V52 < 1114.0) ➡ Trees : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}⟨G⟩(min[V29] ≥ 1347.0) ∧ [G](min[V68] ≥ 563.0) ∧ {2}(V923 ≥ 1714.0) ∧ (V52 < 1114.0) ➡ Bare Soil : (ninstances = 2, confidence = 0.5, coverage = 0.06)

▣ {1}([G](min[V5] ≥ 516.0)) ∧ {2}(V917 ≥ 1490.0) ∧ (V923 < 1714.0) ∧ (V106 < 1346.0) ➡ Gravel : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}([G](min[V1] < 2642.0)) ∧ {2}(V248 ≥ 1882.0) ➡ Bare Soil : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}([G](min[V48] < 2114.0)) ∧ {2}(V742 ≥ 1915.0) ∧ (V248 < 1882.0) ➡ Self-Blocking Bricks : (ninstances = 2, confidence = 0.5, coverage = 0.06)

▣ {1}([G](min[V48] < 2114.0)) ∧ {2}(V132 ≥ 1394.0) ∧ (V248 < 1882.0) ∧ (V742 < 1915.0) ➡ Self-Blocking Bricks : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}([G](min[V48] < 2114.0)) ∧ {2}(V52 ≥ 1114.0) ∧ (V248 < 1882.0) ∧ (V742 < 1915.0) ∧ (V132 < 1394.0) ➡ Gravel : (ninstances = 2, confidence = 0.5, coverage = 0.06)

▣ {1}[G](min[V29] < 1347.0) ∧ [G](min[V37] < 891.0) ∧ {2}(V540 ≥ 428.0) ∧ (V34 < 1009.0) ∧ (V662 < 1908.0) ➡ Meadows : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}[G](min[V38] < 2424.0) ∧ [G](min[V48] < 2114.0) ∧ {2}(V743 ≥ 1919.0) ➡ Self-Blocking Bricks : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V917 ≥ 1401.0) ∧ (V860 < 1700.0) ∧ (V142 < 1281.0) ➡ Gravel : (ninstances = 2, confidence = 0.5, coverage = 0.06)

▣ {1}(⟨G⟩(min[V29] ≥ 1348.0)) ∧ {2}(V477 ≥ 482.0) ∧ (V52 < 1114.0) ➡ Bare Soil : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}([G](min[V29] < 1348.0)) ∧ {2}(V477 ≥ 482.0) ∧ (V52 < 1114.0) ∧ (V762 < 2450.0) ➡ Bare Soil : (ninstances = 2, confidence = 0.0, coverage = 0.06)

▣ {2}(V14 ≥ 1205.0) ∧ (V239 < 1857.0) ∧ (V170 < 1534.0) ➡ Gravel : (ninstances = 2, confidence = 0.5, coverage = 0.06)

▣ {2}(V528 ≥ 1925.0) ∧ (V239 < 1857.0) ∧ (V170 < 1534.0) ∧ (V14 < 1205.0) ➡ Gravel : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V923 < 1714.0) ∧ (V593 < 1343.0) ➡ Asphalt : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}(⟨G⟩(min[V29] ≥ 1347.0)) ∧ {2}(V540 ≥ 428.0) ∧ (V52 < 1114.0) ➡ Bare Soil : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V845 ≥ 1661.0) ∧ (V923 < 1649.0) ➡ Gravel : (ninstances = 2, confidence = 1.0, coverage = 0.06)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V566 ≥ 1385.0) ∧ (V923 < 1649.0) ∧ (V845 < 1661.0) ∧ (V175 < 1261.0) ➡ Asphalt : (ninstances = 2, confidence = 0.0, coverage = 0.06)

▣ {1}[G](min[V47] < 2103.0) ∧ [G](min[V2] ≥ 1021.0) ∧ {2}(V52 ≥ 1122.0) ∧ (V248 < 1882.0) ∧ (V178 < 1476.0) ➡ Gravel : (ninstances = 1, confidence = 1.0, coverage = 0.03)

▣ {1}[G](min[V68] ≥ 563.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V63 ≥ 579.0) ∧ (V52 < 1122.0) ∧ (V662 < 1915.0) ➡ Meadows : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V917 ≥ 1490.0) ∧ (V923 < 1714.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {1}⟨G⟩(min[V3] < 754.0) ∧ [G](min[V6] ≥ 567.0) ∧ {2}(V593 ≥ 1343.0) ∧ (V923 < 1714.0) ∧ (V917 < 1490.0) ➡ Asphalt : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {1}[G](min[V44] … 543.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V662 ≥ 1908.0) ∧

(V44 < 1109.0) ➡ Meadows : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}[G](min[V44] ≥ 543.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V923 ≥ 1714.0) ∧ (V44 < 1109.0) ∧ (V662 < 1908.0) ➡ Bare Soil : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V667 ≥ 1686.0) ∧ (V923 < 1714.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V593 ≥ 1343.0) ∧ (V923 < 1714.0) ∧ (V667 < 1686.0) ∧ (V201 < 1266.0) ➡ Asphalt : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}⟨G⟩(min[V38] ≥ 2267.0) ∧ [G]((min[V38] ≥ 2267.0) → (min[V1] < 2642.0)) ∧ {2}(V52 ≥ 1114.0) ➡ Bare Soil : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}[G](min[V38] < 2267.0) ∧ [G](min[V4] ≥ 1185.0) ∧ {2}(V167 ≥ 1475.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}([G](min[V5] ≥ 516.0)) ∧ {2}(V593 ≥ 1343.0) ∧ (V923 < 1714.0) ∧ (V917 < 1490.0) ➡ Bitumen : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}(⟨G⟩(min[V48] ≥ 2114.0)) ∧ {2}(V52 ≥ 1114.0) ∧ (V248 < 1882.0) ➡ Gravel : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V876 ≥ 1600.0) ∧ (V923 < 1714.0) ➡ Gravel : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V593 ≥ 1343.0) ∧ (V923 < 1714.0) ∧ (V876 < 1600.0) ∧ (V155 < 1168.0) ➡ Asphalt : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}([G](min[V1] < 2642.0)) ∧ {2}(V248 ≥ 1882.0) ➡ Bare Soil : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}(⟨G⟩(min[V48] ≥ 2114.0)) ∧ {2}(V34 ≥ 1009.0) ∧ (V248 < 1882.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}([G](min[V48] < 2114.0)) ∧ {2}(V34 ≥ 1009.0) ∧ (V248 < 1882.0) ∧ (V197 < 1459.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}(⟨G⟩(min[V29] ≥ 1347.0)) ∧ {2}(V540 ≥ 428.0) ∧ (V34 < 1009.0) ➡ Bare Soil : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}([G](min[V29] < 1347.0)) ∧ {2}(V662 ≥ 1908.0) ∧ (V34 < 1009.0) ➡ Meadows : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}⟨G⟩(min[V37] ≥ 891.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V540 ≥ 428.0) ∧ (V34 < 1009.0) ∧ (V662 < 1908.0) ➡ Bare Soil : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}([G](min[V72] ≥ 1534.0)) ∧ {2}(V923 ≥ 1715.0) ∧ (V34 < 1009.0) ∧ (V540 < 428.0) ➡ Meadows : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}([G](min[V6] ≥ 597.0)) ∧ {2}(V876 ≥ 1600.0) ∧ (V923 < 1715.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}[G](min[V6] ≥ 597.0) ∧ [G](min[V18] < 1318.0) ∧ {2}(V593 ≥ 1343.0) ∧ (V923 < 1715.0) ∧ (V876 < 1600.0) ➡ Asphalt : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}⟨G⟩(min[V48] ≥ 2114.0) ∧ [G](min[V38] < 2424.0) ∧ {2}(V52 ≥ 1114.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)
▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V593 ≥ 1343.0) ∧ (V860 < 1700.0) ∧ (V142 < 1281.0) ∧ (V917 < 1401.0) ➡ Bitumen : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}([G](min[V1] < 2642.0)) ∧ {2}(V248 ≥ 1882.0) ➡ Bare Soil : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}(⟨G⟩(min[V29] ≥ 1347.0)) ∧ {2}(V540 ≥ 428.0) ∧ (V52 < 1114.0) ➡ Bare Soil : (ninstances = 1, confidence = 1.0, coverage = 0.03)
▣ {1}⟨G⟩(min[V85] < 1923.0) ∧ [G](min[V29] < 1347.0) ∧ {2}(V540 ≥ 428.0) ∧ (V52 < 1114.0) ➡ Bare Soil : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {1}[G](min[V29] < 1347.0) ∧ [G](min[V85] ≥ 1923.0) ∧ {2}(V6 ≥ 969.0) ∧ (V52 < 1114.0) ➡ Bare Soil : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {1}([G](min[V72] ≥ 1534.0)) ∧ {2}(V923 ≥ 1714.0) ∧ (V52 < 1114.0) ∧ (V540 < 428.0) ➡ Meadows : (ninstances = 1, confidence = 1.0, coverage = 0.03)

▣ {1}([G](min[V6] ≥ 567.0)) ∧ {2}(V415 ≥ 1666.0) ∧ (V923 < 1714.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {1}([G](min[V4] ≥ 496.0)) ∧ {2}(V876 ≥ 1600.0) ∧ (V923 < 1714.0) ➡ Gravel : (ninstances = 1, confidence = 1.0, coverage = 0.03)

▣ {1}([G](min[V4] ≥ 496.0)) ∧ {2}(V575 ≥ 1369.0) ∧ (V923 < 1714.0) ∧ (V876 < 1600.0) ∧ (V155 < 1168.0) ➡ Asphalt : (ninstances = 1, confidence = 1.0, coverage = 0.03)

▣ {1}([G](min[V1] < 2642.0)) ∧ {2}(V239 ≥ 1857.0) ➡ Bare Soil : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {2}(V52 ≥ 1114.0) ∧ (V239 < 1857.0) ∧ (V170 < 1534.0) ∧ (V14 < 1205.0) ∧ (V528 < 1925.0) ➡ Self-Blocking Bricks : (ninstances = 1, confidence = 1.0, coverage = 0.03)

▣ {1}⟨G⟩(min[V102] < 2299.0) ∧ [G](min[V28] < 1302.0) ∧ {2}(V540 ≥ 428.0) ∧ (V52 < 1114.0) ➡ Meadows : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {1}([G](min[V1] < 2642.0)) ∧ {2}(V248 ≥ 1882.0) ➡ Bare Soil : (ninstances = 1, confidence = 1.0, coverage = 0.03)

▣ {2}(V52 ≥ 1114.0) ∧ (V248 < 1882.0) ∧ (V141 < 1464.0) ➡ Gravel : (ninstances = 1, confidence = 0.0, coverage = 0.03)

▣ {1}[G](min[V29] < 1347.0) ∧ [G](min[V102] ≥ 2299.0) ∧ {2}(V540 ≥ 428.0) ∧ (V52 < 1114.0) ➡ Meadows : (ninstances = 1, confidence = 1.0, coverage = 0.03)