

```

library(flexsurv)

## Loading required package: survival
## Loading required package: splines

library(boot)

##
## Attaching package: 'boot'
##
## The following object is masked from 'package:survival':
##
## aml

library(randomForestSRC)

## Loading required package: parallel
##
## randomForestSRC 1.5.5
##
## Type rfsrc.news() to see new features, changes, and bug fixes.
##

library(timeROC)

## Loading required package: pec
## Loading required package: mvtnorm
## Loading required package: timereg

library(risksetROC)

## Loading required package: MASS

library(ggplot2)

## Loading required package: methods

library(RColorBrewer)
pal = brewer.pal(4, "Dark2")
names(pal) = c("gg", "km0", "mskcc.pre", "mskcc.post")

```

## 1 Preparation

Construct a \*preoperative\* function based on the Brennan nomogram. The preoperative nature will mean that most prognostic components will need to be marginalized out.

So the preoperative MSKCC score would be:

$$S = 1.4 + 6.1 + 0.8 + 18.2 + 18.9 + 15 + 9 + 15 * Back.pain + 3 * Weight.Loss + -2/15 * Age + 12 + 3 [Sex = M] + 51 [Hemoglobin < 12] \quad (1)$$

```

fit.mskcc = list(
  inputs = list(
    History.Diagnosis.AgeAt = list(
      margins = data.frame(value = 65, fraction = 1),
      scorefunc = function(x) { x = x; -2/15*pmin(pmax(x, 0), 90) + 12 } ),
    Patient.Sex = list(

```

Variable	Preoperative?	Available?	Marginals
Age	Yes	Yes	Linear. 90 =>0, 30 =>8. Therefore $f(x) = -2/15(x - 90) = -2/15x + 12$
Sex	Yes	Yes	Male risk delta 3
Portal Vein	NO		14.4% YES, risk delta 10, marginal 1.4
Splenectomy	NO		9.9% YES, risk delta 62, marginal 6.1
Margin of resection	NO		20.7% POS, risk delta 4, marginal 0.8
Head.vs.Other	Yes	Yes	Head risk delta 51
Differentiation	NO		14.2% Well, risk delta 0, marginal 0 56.4% Mod, risk delta 14, marginal 7.9 29.5% Poor, risk delta 35, marginal 10.3. Overall marginal 18.2
Posterior.margin	NO		86.0% POS, risk delta 22, marginal 18.9
Numb.pos.nodes	NO		Mean 2.1, approx marginal 15
Numb.neg.nodes	NO		Mean 16.9, approx marginal 9
Back.pain	Yes	NO	13.7% YES, risk delta 15, marginal 2.0
T.stage	Yes	Yes	
Weight Loss	Yes	NO	53.7% YES, risk delta 3, marginal 1.6
Max.path.axis	Yes	Yes	

```

margins = data.frame(value = c("M", "F"), fraction = c(0.501, 1-0.501)),
scorefunc = function(x) { 3*I(x == "M") }},
Portal.Vein = list(
  margins = data.frame(value = c(TRUE, FALSE), fraction = c(0.144, 1-0.144)),
  scorefunc = function(x) { 10*I(x == TRUE) }},
Splenectomy = list(
  margins = data.frame(value = c(TRUE, FALSE), fraction = c(0.099, 1-0.099)),
  scorefunc = function(x) { 62*I(x == TRUE) }},
Treat.MarginPositive = list(
  margins = data.frame(value = c(TRUE, FALSE), fraction = c(0.207, 1-0.207)),
  scorefunc = function(x) { 4*I(x == TRUE) }},
Path.LocationBody = list(
  margins = data.frame(value = c(FALSE, TRUE), fraction = c(0.894, 1-0.894)),
  scorefunc = function(x) { 51*I(x == TRUE) }},
Path.Differentiation = list(
  margins = data.frame(value = c("1", "2", "3", "4"), fraction = c(0.142, 0.564, 1-0.142-0.564)),
  scorefunc = function(x) { 14*I(x == "2") + 35*I(x == "3") + 35*I(x == "4") }},
Posterior.Margin = list(
  margins = data.frame(value = c(TRUE, FALSE), fraction = c(0.86, 1-0.86)),
  scorefunc = function(x) { 22*I(x == TRUE) }},
Path.LN.Involved = list(
  margins = data.frame(value = 2.1, fraction = 1),
  scorefunc = function(x) {
    x = pmin(40, pmax(x, 0))
    fitfun = splinefun(c(0, 1, 2, 3, 4, 10, 15, 20, 25, 30, 35, 40), c(0, 14.56, 24.56, 35.56, 46.56, 57.56, 68.56, 79.56, 90.56, 101.56, 112.56, 123.56, 134.56, 145.56, 156.56, 167.56, 178.56, 189.56, 200.56, 211.56, 222.56, 233.56, 244.56, 255.56, 266.56, 277.56, 288.56, 299.56, 310.56, 321.56, 332.56, 343.56, 354.56, 365.56, 376.56, 387.56, 398.56, 409.56, 420.56, 431.56, 442.56, 453.56, 464.56, 475.56, 486.56, 497.56, 508.56, 519.56, 530.56, 541.56, 552.56, 563.56, 574.56, 585.56, 596.56, 607.56, 618.56, 629.56, 640.56, 651.56, 662.56, 673.56, 684.56, 695.56, 706.56, 717.56, 728.56, 739.56, 750.56, 761.56, 772.56, 783.56, 794.56, 805.56, 816.56, 827.56, 838.56, 849.56, 860.56, 871.56, 882.56, 893.56, 904.56, 915.56, 926.56, 937.56, 948.56, 959.56, 970.56, 981.56, 992.56, 1003.56, 1014.56, 1025.56, 1036.56, 1047.56, 1058.56, 1069.56, 1080.56, 1091.56, 1102.56, 1113.56, 1124.56, 1135.56, 1146.56, 1157.56, 1168.56, 1179.56, 1190.56, 1201.56, 1212.56, 1223.56, 1234.56, 1245.56, 1256.56, 1267.56, 1278.56, 1289.56, 1300.56, 1311.56, 1322.56, 1333.56, 1344.56, 1355.56, 1366.56, 1377.56, 1388.56, 1399.56, 1410.56, 1421.56, 1432.56, 1443.56, 1454.56, 1465.56, 1476.56, 1487.56, 1498.56, 1509.56, 1520.56, 1531.56, 1542.56, 1553.56, 1564.56, 1575.56, 1586.56, 1597.56, 1608.56, 1619.56, 1630.56, 1641.56, 1652.56, 1663.56, 1674.56, 1685.56, 1696.56, 1707.56, 1718.56, 1729.56, 1740.56, 1751.56, 1762.56, 1773.56, 1784.56, 1795.56, 1806.56, 1817.56, 1828.56, 1839.56, 1850.56, 1861.56, 1872.56, 1883.56, 1894.56, 1905.56, 1916.56, 1927.56, 1938.56, 1949.56, 1960.56, 1971.56, 1982.56, 1993.56, 2004.56, 2015.56, 2026.56, 2037.56, 2048.56, 2059.56, 2070.56, 2081.56, 2092.56, 2103.56, 2114.56, 2125.56, 2136.56, 2147.56, 2158.56, 2169.56, 2180.56, 2191.56, 2202.56, 2213.56, 2224.56, 2235.56, 2246.56, 2257.56, 2268.56, 2279.56, 2290.56, 2301.56, 2312.56, 2323.56, 2334.56, 2345.56, 2356.56, 2367.56, 2378.56, 2389.56, 2400.56, 2411.56, 2422.56, 2433.56, 2444.56, 2455.56, 2466.56, 2477.56, 2488.56, 2499.56, 2510.56, 2521.56, 2532.56, 2543.56, 2554.56, 2565.56, 2576.56, 2587.56, 2598.56, 2609.56, 2620.56, 2631.56, 2642.56, 2653.56, 2664.56, 2675.56, 2686.56, 2697.56, 2708.56, 2719.56, 2730.56, 2741.56, 2752.56, 2763.56, 2774.56, 2785.56, 2796.56, 2807.56, 2818.56, 2829.56, 2840.56, 2851.56, 2862.56, 2873.56, 2884.56, 2895.56, 2906.56, 2917.56, 2928.56, 2939.56, 2950.56, 2961.56, 2972.56, 2983.56, 2994.56, 3005.56, 3016.56, 3027.56, 3038.56, 3049.56, 3060.56, 3071.56, 3082.56, 3093.56, 3104.56, 3115.56, 3126.56, 3137.56, 3148.56, 3159.56, 3170.56, 3181.56, 3192.56, 3203.56, 3214.56, 3225.56, 3236.56, 3247.56, 3258.56, 3269.56, 3280.56, 3291.56, 3302.56, 3313.56, 3324.56, 3335.56, 3346.56, 3357.56, 3368.56, 3379.56, 3390.56, 3401.56, 3412.56, 3423.56, 3434.56, 3445.56, 3456.56, 3467.56, 3478.56, 3489.56, 3500.56, 3511.56, 3522.56, 3533.56, 3544.56, 3555.56, 3566.56, 3577.56, 3588.56, 3599.56, 3610.56, 3621.56, 3632.56, 3643.56, 3654.56, 3665.56, 3676.56, 3687.56, 3698.56, 3709.56, 3720.56, 3731.56, 3742.56, 3753.56, 3764.56, 3775.56, 3786.56, 3797.56, 3808.56, 3819.56, 3830.56, 3841.56, 3852.56, 3863.56, 3874.56, 3885.56, 3896.56, 3907.56, 3918.56, 3929.56, 3940.56, 3951.56, 3962.56, 3973.56, 3984.56, 3995.56, 4006.56, 4017.56, 4028.56, 4039.56, 4050.56, 4061.56, 4072.56, 4083.56, 4094.56, 4105.56, 4116.56, 4127.56, 4138.56, 4149.56, 4160.56, 4171.56, 4182.56, 4193.56, 4204.56, 4215.56, 4226.56, 4237.56, 4248.56, 4259.56, 4270.56, 4281.56, 4292.56, 4303.56, 4314.56, 4325.56, 4336.56, 4347.56, 4358.56, 4369.56, 4380.56, 4391.56, 4402.56, 4413.56, 4424.56, 4435.56, 4446.56, 4457.56, 4468.56, 4479.56, 4490.56, 4501.56, 4512.56, 4523.56, 4534.56, 4545.56, 4556.56, 4567.56, 4578.56, 4589.56, 4600.56, 4611.56, 4622.56, 4633.56, 4644.56, 4655.56, 4666.56, 4677.56, 4688.56, 4699.56, 4710.56, 4721.56, 4732.56, 4743.56, 4754.56, 4765.56, 4776.56, 4787.56, 4798.56, 4809.56, 4820.56, 4831.56, 4842.56, 4853.56, 4864.56, 4875.56, 4886.56, 4897.56, 4908.56, 4919.56, 4930.56, 4941.56, 4952.56, 4963.56, 4974.56, 4985.56, 4996.56, 5007.56, 5018.56, 5029.56, 5040.56, 5051.56, 5062.56, 5073.56, 5084.56, 5095.56, 5106.56, 5117.56, 5128.56, 5139.56, 5150.56, 5161.56, 5172.56, 5183.56, 5194.56, 5205.56, 5216.56, 5227.56, 5238.56, 5249.56, 5260.56, 5271.56, 5282.56, 5293.56, 5304.56, 5315.56, 5326.56, 5337.56, 5348.56, 5359.56, 5370.56, 5381.56, 5392.56, 5403.56, 5414.56, 5425.56, 5436.56, 5447.56, 5458.56, 5469.56, 5480.56, 5491.56, 5502.56, 5513.56, 5524.56, 5535.56, 5546.56, 5557.56, 5568.56, 5579.56, 5590.56, 5601.56, 5612.56, 5623.56, 5634.56, 5645.56, 5656.56, 5667.56, 5678.56, 5689.56, 5700.56, 5711.56, 5722.56, 5733.56, 5744.56, 5755.56, 5766.56, 5777.56, 5788.56, 5799.56, 5810.56, 5821.56, 5832.56, 5843.56, 5854.56, 5865.56, 5876.56, 5887.56, 5898.56, 5909.56, 5920.56, 5931.56, 5942.56, 5953.56, 5964.56, 5975.56, 5986.56, 5997.56, 6008.56, 6019.56, 6030.56, 6041.56, 6052.56, 6063.56, 6074.56, 6085.56, 6096.56, 6107.56, 6118.56, 6129.56, 6140.56, 6151.56, 6162.56, 6173.56, 6184.56, 6195.56, 6206.56, 6217.56, 6228.56, 6239.56, 6250.56, 6261.56, 6272.56, 6283.56, 6294.56, 6305.56, 6316.56, 6327.56, 6338.56, 6349.56, 6360.56, 6371.56, 6382.56, 6393.56, 6404.56, 6415.56, 6426.56, 6437.56, 6448.56, 6459.56, 6470.56, 6481.56, 6492.56, 6503.56, 6514.56, 6525.56, 6536.56, 6547.56, 6558.56, 6569.56, 6580.56, 6591.56, 6602.56, 6613.56, 6624.56, 6635.56, 6646.56, 6657.56, 6668.56, 6679.56, 6690.56, 6701.56, 6712.56, 6723.56, 6734.56, 6745.56, 6756.56, 6767.56, 6778.56, 6789.56, 6800.56, 6811.56, 6822.56, 6833.56, 6844.56, 6855.56, 6866.56, 6877.56, 6888.56, 6899.56, 6910.56, 6921.56, 6932.56, 6943.56, 6954.56, 6965.56, 6976.56, 6987.56, 6998.56, 7009.56, 7020.56, 7031.56, 7042.56, 7053.56, 7064.56, 7075.56, 7086.56, 7097.56, 7108.56, 7119.56, 7130.56, 7141.56, 7152.56, 7163.56, 7174.56, 7185.56, 7196.56, 7207.56, 7218.56, 7229.56, 7240.56, 7251.56, 7262.56, 7273.56, 7284.56, 7295.56, 7306.56, 7317.56, 7328.56, 7339.56, 7350.56, 7361.56, 7372.56, 7383.56, 7394.56, 7405.56, 7416.56, 7427.56, 7438.56, 7449.56, 7460.56, 7471.56, 7482.56, 7493.56, 7504.56, 7515.56, 7526.56, 7537.56, 7548.56, 7559.56, 7570.56, 7581.56, 7592.56, 7603.56, 7614.56, 7625.56, 7636.56, 7647.56, 7658.56, 7669.56, 7680.56, 7691.56, 7702.56, 7713.56, 7724.56, 7735.56, 7746.56, 7757.56, 7768.56, 7779.56, 7790.56, 7801.56, 7812.56, 7823.56, 7834.56, 7845.56, 7856.56, 7867.56, 7878.56, 7889.56, 7900.56, 7911.56, 7922.56, 7933.56, 7944.56, 7955.56, 7966.56, 7977.56, 7988.56, 7999.56, 8010.56, 8021.56, 8032.56, 8043.56, 8054.56, 8065.56, 8076.56, 8087.56, 8098.56, 8109.56, 8120.56, 8131.56, 8142.56, 8153.56, 8164.56, 8175.56, 8186.56, 8197.56, 8208.56, 8219.56, 8230.56, 8241.56, 8252.56, 8263.56, 8274.56, 8285.56, 8296.56, 8307.56, 8318.56, 8329.56, 8340.56, 8351.56, 8362.56, 8373.56, 8384.56, 8395.56, 8406.56, 8417.56, 8428.56, 8439.56, 8450.56, 8461.56, 8472.56, 8483.56, 8494.56, 8505.56, 8516.56, 8527.56, 8538.56, 8549.56, 8560.56, 8571.56, 8582.56, 8593.56, 8604.56, 8615.56, 8626.56, 8637.56, 8648.56, 8659.56, 8670.56, 8681.56, 8692.56, 8703.56, 8714.56, 8725.56, 8736.56, 8747.56, 8758.56, 8769.56, 8780.56, 8791.56, 8802.56, 8813.56, 8824.56, 8835.56, 8846.56, 8857.56, 8868.56, 8879.56, 8890.56, 8901.56, 8912.56, 8923.56, 8934.56, 8945.56, 8956.56, 8967.56, 8978.56, 8989.56, 9000.56, 9011.56, 9022.56, 9033.56, 9044.56, 9055.56, 9066.56, 9077.56, 9088.56, 9099.56, 9110.56, 9121.56, 9132.56, 9143.56, 9154.56, 9165.56, 9176.56, 9187.56, 9198.56, 9209.56, 9220.56, 9231.56, 9242.56, 9253.56, 9264.56, 9275.56, 9286.56, 9297.56, 9308.56, 9319.56, 9330.56, 9341.56, 9352.56, 9363.56, 9374.56, 9385.56, 9396.56, 9407.56, 9418.56, 9429.56, 9440.56, 9451.56, 9462.56, 9473.56, 9484.56, 9495.56, 9506.56, 9517.56, 9528.56, 9539.56, 9550.56, 9561.56, 9572.56, 9583.56, 9594.56, 9605.56, 9616.56, 9627.56, 9638.56, 9649.56, 9660.56, 9671.56, 9682.56, 9693.56, 9704.56, 9715.56, 9726.56, 9737.56, 9748.56, 9759.56, 9770.56, 9781.56, 9792.56, 9803.56, 9814.56, 9825.56, 9836.56, 9847.56, 9858.56, 9869.56, 9880.56, 9891.56, 9902.56, 9913.56, 9924.56, 9935.56, 9946.56, 9957.56, 9968.56, 9979.56, 9990.56, 10001.56, 10012.56, 10023.56, 10034.56, 10045.56, 10056.56, 10067.56, 10078.56, 10089.56, 10100.56, 10111.56, 10122.56, 10133.56, 10144.56, 10155.56, 10166.56, 10177.56, 10188.56, 10199.56, 10210.56, 10221.56, 10232.56, 10243.56, 10254.56, 10265.56, 10276.56, 10287.56, 10298.56, 10309.56, 10320.56, 10331.56, 10342.56, 10353.56, 10364.56, 10375.56, 10386.56, 10397.56, 10408.56, 10419.56, 10430.56, 10441.56, 10452.56, 10463.56, 10474.56, 10485.56, 10496.56, 10507.56, 10518.56, 10529.56, 10540.56, 10551.56, 10562.56, 10573.56, 10584.56, 10595.56, 10606.56, 10617.56, 10628.56, 10639.56, 10650.56, 10661.56, 10672.56, 10683.56, 10694.56, 10705.56, 10716.56, 10727.56, 10738.56, 10749.56, 10760.56, 10771.56, 10782.56, 10793.56, 10804.56, 10815.56, 10826.56, 10837.56, 10848.56, 10859.56, 10870.56, 10881.56, 10892.56, 10903.56, 10914.56, 10925.56, 10936.56, 10947.56, 10958.56, 10969.56, 10980.56, 10991.56, 11002.56, 11013.56, 11024.56, 11035.56, 11046.56, 11057.56, 11068.56, 11079.56, 11090.56, 11101.56, 11112.56, 11123.56, 11134.56, 11145.56, 11156.56, 11167.56, 11178.56, 11189.56, 11200.56, 11211.56, 11222.56, 11233.56, 11244.56, 11255.56, 11266.56, 11277.56, 11288.56, 11299.56, 11310.56, 11321.56, 11332.56, 11343.56, 11354.56, 11365.56, 11376.56, 11387.56, 11398.56, 11409.56, 11420.56, 11431.56, 11442.56, 11453.56, 11464.56, 11475.56, 11486.56, 11497.56, 11508.56, 11519.56, 11530.56, 11541.56, 11552.56, 11563.56, 11574.56, 11585.56, 11596.56, 11607.56, 11618.56, 11629.56, 11640.56, 11651.56, 11662.56, 11673.56, 11684.56, 11695.56, 11706.56, 11717.56, 11728.56, 11739.56, 11750.56, 11761.56, 11772.56, 11783.56, 11794.56, 11805.56, 11816.56, 11827.56, 11838.56, 11849.56, 11860.56, 11871.56, 11882.56, 11893.56, 11904.56, 11915.56, 11926.56, 11937.56, 11948.56, 11959.56, 11970.56, 11981.56, 11992.56, 12003.56, 12014.56, 12025.56, 12036.56, 12047.56, 12058.56, 12069.56, 12080.56, 12091.56, 12102.56, 12113.56, 12124.56, 12135.56, 12146.56, 12157.56, 12168.56, 12179.56, 12190.56, 12201.56, 12212.56, 12223.56, 12234.56, 12245.56, 12256.56, 12267.56, 12278.56, 12289.56, 12300.56, 12311.56, 12322.56, 12333.56, 12344.56, 12355.56, 12366.56, 12377.56, 12388.56, 12399.56, 12410.56, 12421.56, 12432.56, 12443.56, 12454.56, 12465.56, 12476.56, 12487.56, 12498.56, 12509.56, 12520.56, 12531.56, 12542.56, 12553.56, 12564.56, 12575.56, 12586.56, 12597.56, 12608.56, 12619.56, 12630.56, 12641.56, 12652.56, 12663.56, 12674.56, 12685.56, 12696.56, 12707.56, 12718.56, 12729.56, 12740.56, 12751.56, 12762.56, 12773.56, 12784.56, 12795.56, 12806.56, 12817.56, 12828.56, 12839.56, 12850.56, 12861.56, 12872.56, 12883.56, 12894.56, 12905.56, 12916.56, 12927.56, 12938.56, 12949.56, 12960.56, 12971.56, 12982.56, 12993.56, 13004.56, 13015.56, 13026.56, 13037.56, 13048.56, 13059.56, 13070.56, 13081.56, 13092.56, 13103.56, 13114.56, 13125.56, 13136.56, 13147.56, 13158.56, 13169.56, 13180.56, 13191.56, 13202.56, 13213.56, 13224.56, 13235.56, 13246.56, 13257.56, 13268.56, 13279.56, 13290.56
```

```

scorefunc = function(x) { 36*I(x == "T1") + 11*I(x == "T34") }),
# The following matches the original Brennan nomogram, but was not used as there are too
# tumours in either the NSWPCN *or* the MSKCC cohorts -- how the T4 coefficient was even
# I'll never know. The T34 coefficient of 11 was arrived at as  $(0.828 \times 10 + (1 - 0.037 - 0.119) \times 10)$ 
# being a frequency-weighted average of the T3 and T4 coefficients.
# margins = data.frame(value = c("T1", "T2", "T3", "T4"), fraction = c(0.037, 0.119, 0.119, 0.725))
# scorefunc = function(x) { 36*I(x == "T1") + 10*I(x == "T3") + 63*I(x == "T4") }),
Weight.loss = list(
  margins = data.frame(value = c(TRUE, FALSE), fraction = c(0.537, 1-0.537)),
  scorefunc = function(x) { 3*I(x == TRUE) }),
Path.Size = list(
  margins = data.frame(),
  scorefunc = function(x) {
    x = pmin(16, pmax(x, 0))
    fitfun = splinefun(c(0, 1, 2, 3, 4, 6, 8, 10, 12, 14, 16), c(0, 29.74, 59.48, 89.22, 118.96, 148.69, 178.43, 208.17, 237.91, 267.65, 297.39))
    fitfun(x)
  }) ),
outputs = list(
  DSS12mo = function(s) {
    x = pmax(50, pmin(350, s))
    fitfun = splinefun(c(79.0323, 115.02, 165.524, 197.278, 221.774, 242.339, 261.081, 280.735, 300.389, 320.043, 339.697, 359.351))
    y = fitfun(x)
    pmax(0, pmin(1, y))
  },
  DSS24mo = function(s) {
    x = pmax(50, pmin(350, s))
    fitfun = splinefun(c(71.1694, 97.7823, 129.536, 153.73, 174.294, 193.347, 211.791, 230.844, 249.897, 268.95, 288.003, 307.056, 326.109, 345.162, 364.215, 383.268, 402.321, 421.374, 440.427, 459.48, 478.533, 497.586, 516.639, 535.692, 554.745, 573.798, 592.851, 611.904, 630.957, 650.01, 669.063, 688.116, 707.169, 726.222, 745.275, 764.328, 783.381, 802.434, 821.487, 840.54, 859.593, 878.646, 897.699, 916.752, 935.805, 954.858, 973.911, 992.964, 1012.017, 1031.07, 1050.123, 1069.176, 1088.229, 1107.282, 1126.335, 1145.388, 1164.441, 1183.494, 1202.547, 1221.6, 1240.653, 1259.706, 1278.759, 1297.812, 1316.865, 1335.918, 1354.971, 1374.024, 1393.077, 1412.13, 1431.183, 1450.236, 1469.289, 1488.342, 1507.395, 1526.448, 1545.501, 1564.554, 1583.607, 1602.66, 1621.713, 1640.766, 1659.819, 1678.872, 1697.925, 1716.978, 1736.031, 1755.084, 1774.137, 1793.19, 1812.243, 1831.296, 1850.349, 1869.402, 1888.455, 1907.508, 1926.561, 1945.614, 1964.667, 1983.72, 2002.773, 2021.826, 2040.879, 2059.932, 2078.985, 2098.038, 2117.091, 2136.144, 2155.197, 2174.25, 2193.303, 2212.356, 2231.409, 2250.462, 2269.515, 2288.568, 2307.621, 2326.674, 2345.727, 2364.78, 2383.833, 2402.886, 2421.939, 2440.992, 2460.045, 2479.098, 2498.151, 2517.204, 2536.257, 2555.31, 2574.363, 2593.416, 2612.469, 2631.522, 2650.575, 2669.628, 2688.681, 2707.734, 2726.787, 2745.84, 2764.893, 2783.946, 2802.999, 2822.052, 2841.105, 2860.158, 2879.211, 2898.264, 2917.317, 2936.37, 2955.423, 2974.476, 2993.529, 3012.582, 3031.635, 3050.688, 3069.741, 3088.794, 3107.847, 3126.9, 3145.953, 3165.006, 3184.059, 3203.112, 3222.165, 3241.218, 3260.271, 3279.324, 3298.377, 3317.43, 3336.483, 3355.536, 3374.589, 3393.642, 3412.695, 3431.748, 3450.801, 3469.854, 3488.907, 3507.96, 3527.013, 3546.066, 3565.119, 3584.172, 3603.225, 3622.278, 3641.331, 3660.384, 3679.437, 3698.49, 3717.543, 3736.596, 3755.649, 3774.702, 3793.755, 3812.808, 3831.861, 3850.914, 3869.967, 3889.02, 3908.073, 3927.126, 3946.179, 3965.232, 3984.285, 4003.338, 4022.391, 4041.444, 4060.497, 4079.55, 4098.603, 4117.656, 4136.709, 4155.762, 4174.815, 4193.868, 4212.921, 4231.974, 4251.027, 4270.08, 4289.133, 4308.186, 4327.239, 4346.292, 4365.345, 4384.398, 4403.451, 4422.504, 4441.557, 4460.61, 4479.663, 4498.716, 4517.769, 4536.822, 4555.875, 4574.928, 4593.981, 4613.034, 4632.087, 4651.14, 4670.193, 4689.246, 4708.299, 4727.352, 4746.405, 4765.458, 4784.511, 4803.564, 4822.617, 4841.67, 4860.723, 4879.776, 4898.829, 4917.882, 4936.935, 4955.988, 4975.041, 4994.094, 5013.147, 5032.2, 5051.253, 5070.306, 5089.359, 5108.412, 5127.465, 5146.518, 5165.571, 5184.624, 5203.677, 5222.73, 5241.783, 5260.836, 5279.889, 5298.942, 5317.995, 5337.048, 5356.101, 5375.154, 5394.207, 5413.26, 5432.313, 5451.366, 5470.419, 5489.472, 5508.525, 5527.578, 5546.631, 5565.684, 5584.737, 5603.79, 5622.843, 5641.896, 5660.949, 5679.999, 5699.05, 5718.101, 5737.152, 5756.203, 5775.254, 5794.305, 5813.356, 5832.407, 5851.458, 5870.509, 5889.56, 5908.611, 5927.662, 5946.713, 5965.764, 5984.815, 6003.866, 6022.917, 6041.968, 6061.019, 6080.07, 6099.121, 6118.172, 6137.223, 6156.274, 6175.325, 6194.376, 6213.427, 6232.478, 6251.529, 6270.58, 6289.631, 6308.682, 6327.733, 6346.784, 6365.835, 6384.886, 6403.937, 6422.988, 6442.039, 6461.09, 6480.141, 6499.192, 6518.243, 6537.294, 6556.345, 6575.396, 6594.447, 6613.498, 6632.549, 6651.6, 6670.651, 6689.702, 6708.753, 6727.804, 6746.855, 6765.906, 6784.957, 6804.008, 6823.059, 6842.11, 6861.161, 6880.212, 6899.263, 6918.314, 6937.365, 6956.416, 6975.467, 6994.518, 7013.569, 7032.62, 7051.671, 7070.722, 7089.773, 7108.824, 7127.875, 7146.926, 7165.977, 7185.028, 7204.079, 7223.13, 7242.181, 7261.232, 7280.283, 7299.334, 7318.385, 7337.436, 7356.487, 7375.538, 7394.589, 7413.64, 7432.691, 7451.742, 7470.793, 7489.844, 7508.895, 7527.946, 7546.997, 7566.048, 7585.099, 7604.15, 7623.201, 7642.252, 7661.303, 7680.354, 7699.405, 7718.456, 7737.507, 7756.558, 7775.609, 7794.66, 7813.711, 7832.762, 7851.813, 7870.864, 7889.915, 7908.966, 7928.017, 7947.068, 7966.119, 7985.17, 8004.221, 8023.272, 8042.323, 8061.374, 8080.425, 8099.476, 8118.527, 8137.578, 8156.629, 8175.68, 8194.731, 8213.782, 8232.833, 8251.884, 8270.935, 8289.986, 8309.037, 8328.088, 8347.139, 8366.19, 8385.241, 8404.292, 8423.343, 8442.394, 8461.445, 8480.496, 8499.547, 8518.598, 8537.649, 8556.7, 8575.751, 8594.802, 8613.853, 8632.904, 8651.955, 8671.006, 8690.057, 8709.108, 8728.159, 8747.21, 8766.261, 8785.312, 8804.363, 8823.414, 8842.465, 8861.516, 8880.567, 8899.618, 8918.669, 8937.72, 8956.771, 8975.822, 8994.873, 9013.924, 9032.975, 9052.026, 9071.077, 9090.128, 9109.179, 9128.23, 9147.281, 9166.332, 9185.383, 9204.434, 9223.485, 9242.536, 9261.587, 9280.638, 9299.689, 9318.74, 9337.791, 9356.842, 9375.893, 9394.944, 9413.995, 9433.046, 9452.097, 9471.148, 9490.199, 9509.25, 9528.301, 9547.352, 9566.403, 9585.454, 9604.505, 9623.556, 9642.607, 9661.658, 9680.709, 9699.76, 9718.811, 9737.862, 9756.913, 9775.964, 9795.015, 9814.066, 9833.117, 9852.168, 9871.219, 9890.27, 9909.321, 9928.372, 9947.423, 9966.474, 9985.525, 10004.576, 10023.627, 10042.678, 10061.729, 10080.78, 10099.831, 10118.882, 10137.933, 10156.984, 10176.035, 10195.086, 10214.137, 10233.188, 10252.239, 10271.29, 10290.341, 10309.392, 10328.443, 10347.494, 10366.545, 10385.596, 10404.647, 10423.698, 10442.749, 10461.8, 10480.851, 10500.902, 10519.953, 10539.004, 10558.055, 10577.106, 10596.157, 10615.208, 10634.259, 10653.31, 10672.361, 10691.412, 10710.463, 10729.514, 10748.565, 10767.616, 10786.667, 10805.718, 10824.769, 10843.82, 10862.871, 10881.922, 10900.973, 10920.024, 10939.075, 10958.126, 10977.177, 10996.228, 11015.279, 11034.33, 11053.381, 11072.432, 11091.483, 11110.534, 11129.585, 11148.636, 11167.687, 11186.738, 11205.789, 11224.84, 11243.891, 11262.942, 11281.993, 11301.044, 11320.095, 11339.146, 11358.197, 11377.248, 11396.299, 11415.35, 11434.401, 11453.452, 11472.503, 11491.554, 11510.605, 11529.656, 11548.707, 11567.758, 11586.809, 11605.86, 11624.911, 11643.962, 11663.013, 11682.064, 11701.115, 11720.166, 11739.217, 11758.268, 11777.319, 11796.37, 11815.421, 11834.472, 11853.523, 11872.574, 11891.625, 11910.676, 11929.727, 11948.778, 11967.829, 11986.88, 12005.931, 12024.982, 12044.033, 12063.084, 12082.135, 12101.186, 12120.237, 12139.288, 12158.339, 12177.39, 12196.441, 12215.492, 12234.543, 12253.594, 12272.645, 12291.696, 12310.747, 12329.798, 12348.849, 12367.9, 12386.951, 12406.002, 12425.053, 12444.104, 12463.155, 12482.206, 12501.257, 12520.308, 12539.359, 12558.41, 12577.461, 12596.512, 12615.563, 12634.614, 12653.665, 12672.716, 12691.767, 12710.818, 12729.869, 12748.92, 12767.971, 12787.022, 12806.073, 12825.124, 12844.175, 12863.226, 12882.277, 12901.328, 12920.379, 12939.43, 12958.481, 12977.532, 12996.583, 13015.634, 13034.685, 13053.736, 13072.787, 13091.838, 13110.889, 13129.94, 13148.991, 13168.042, 13187.093, 13206.144, 13225.195, 13244.246, 13263.297, 13282.348, 13301.399, 13320.45, 13339.501, 13358.552, 13377.603, 13396.654, 13415.705, 13434.756, 13453.807, 13472.858, 13491.909, 13510.96, 13530.011, 13549.062, 13568.113, 13587.164, 13606.215, 13625.266, 13644.317, 13663.368, 13682.419, 13701.47, 13720.521, 13739.572, 13758.623, 13777.674, 13796.725, 13815.776, 13834.827, 13853.878, 13872.929, 13891.98, 13911.031, 13930.082, 13949.133, 13968.184, 13987.235, 14006.286, 14025.337, 14044.388, 14063.439, 14082.49, 14101.541, 14120.592, 14139.643, 14158.694, 14177.745, 14196.796, 14215.847, 14234.898, 14253.949, 14272.999, 14292.05, 14311.101, 14330.152, 14349.203, 14368.254, 14387.305, 14406.356, 14425.407, 14444.458, 14463.509, 14482.56, 14501.611, 14520.662, 14539.713, 14558.764, 14577.815, 14596.866, 14615.917, 14634.968, 14654.019, 14673.07, 14692.121, 14711.172, 14730.223, 14749.274, 14768.325, 14787.376, 14806.427, 14825.478, 14844.529, 14863.58, 14882.631, 14901.682, 14920.733, 14939.784, 14958.835, 14977.886, 14996.937, 15015.988, 15035.039, 15054.09, 15073.141, 15092.192, 15111.243, 15130.294, 15149.345, 15168.396, 15187.447, 15206.498, 15225.549, 15244.6, 15263.651, 15282.702, 15301.753, 15320.804, 15339.855, 15358.906, 15377.957, 15397.008, 15416.059, 15435.11, 15454.161, 15473.212, 15492.263, 15511.314, 15530.365, 15549.416, 15568.467, 15587.518, 15606.569, 15625.62, 15644.671, 15663.722, 15682.773, 15701.824, 15720.875, 15739.926, 15758.977, 15778.028, 15797.079, 15816.13, 15835.181, 15854.232, 15873.283, 15892.334, 15911.385, 15930.436, 15949.487, 15968.538, 15987.589, 16006.64, 16025.691, 16044.742, 16063.793, 16082.844, 16101.895, 16120.946, 16139.997, 16159.048, 16178.099, 16197.15, 16216.201, 16235.252, 16254.303, 16273.354, 16292.405, 16311.456, 16330.507, 16349.558, 16368.609, 16387.66, 16406.711, 16425.762, 16444.813, 16463.864, 16482.915, 16501.966, 16521.017, 16540.068, 16559.119, 16578.17, 16597.221, 16616.272, 16635.323, 16654.374, 16673.425, 16692.476, 16711.527, 16730.578, 16749.629, 16768.68, 16787.731, 16806.782, 16825.833, 16844.884, 16863.935, 16882.986, 16902.037, 16921.088, 16940.139, 16959.19, 16978.241, 16997.292, 17016.343, 17035.394, 17054.445, 17073.496, 17092.547, 17111.598, 17130.649, 17149.7, 17168.751, 17187.802, 17206.853, 17225.904, 17244.955, 17264.006, 17283.057, 17302.108, 17321.159, 17340.21, 17359.261, 17378.312, 17397.363, 17416.414, 17435.465, 17454.516, 17473.567, 17492.618, 17511.669, 17530.72, 17549.771, 17568.822, 17587.873, 17606.924, 17625.975, 17645.026, 17664.077, 17683.128, 17702.179, 17721.23, 17740.281, 17759.332, 17778.383, 17797.434, 17816.485, 17835.536, 17854.587, 17873.638, 17892.689, 17911.74, 17930.791, 17949.842, 17968.893, 17987.944, 18006.995, 18026.046, 18045.097, 18064.148, 18083.199, 18102.25, 18121.301, 18140.352, 18159.403, 18178.454, 18197.505, 18216.556, 18235.607, 18254.658, 18273.709, 18292.76, 18311.811, 18330.862, 18349.913, 18368.964, 18388.015, 18407.066, 18426.117, 18445.168, 18464.219, 18483.27, 18502.321, 18521.372, 18540.423, 18559.474, 18578.525, 18597.576, 18616.627, 18635.678, 18654.729, 18673.78, 18692.831, 18711.882, 18730.933, 18750.084, 18769.135, 18788.186, 18807.237, 18826.288, 18845.339, 18864.39, 18883.441, 18902.492, 18921.543, 18940.594, 18959.645, 18978.696, 18997.747, 19016.798, 19035.849, 19054.9, 19073.951, 19093.002, 19112.053, 19131.104, 19150.155, 19169.206, 19188.257, 19207.308, 19226.359, 19245.41, 19264.461, 19283.512, 19302.563, 19321.614, 19340.665, 19359.716, 19378.767, 19397.818, 19416.869, 19435.92, 19454.971, 19474.022, 19493.073, 19512.124, 19531.175, 19550.226, 19569.277, 19588.328, 19607.379, 19626.43, 19645.481, 19664.532, 19683.583, 19702.634, 19721.685, 19740.736, 19759.787, 19778.838, 19797.889, 19816.94, 19835.991, 19855.042, 19874.093, 19893.144, 19912.195, 19931.246, 19950.297, 19969.348, 19988.399, 20007.45, 20026.501, 20045.552, 20064.603, 20083.654, 20102.705, 20121.756, 20140.807, 20159.858, 20178.909, 20197.96, 20217.011, 20236.062, 20255.113, 20274.164, 20293.215, 20312.266, 20331.317, 20350.368, 20369.419, 20388.47, 20407.521, 20426.572, 20445.623, 20464.674, 20483.725, 20502.776, 20521.827, 20540.878, 20559.929, 20578.98, 20598.031, 20617.082, 20636.133, 20655.184, 20674.235, 20693.286, 20712.337, 20731.388, 20750.439, 20769.49, 20788.541, 20807.592, 20826.643, 20845.694, 20864.745, 20883.796, 20902.847, 20921.898, 20940.949, 20960.0, 20979.051, 20998.102, 21017.153, 21036.204, 21055.255, 21074.306, 21093.357, 21112.408, 21131.459, 21150.51, 21169.561, 21188.612, 21207.663, 21226.714, 21245.765, 21264.816, 21283.867, 21302.918, 21321.969, 21341.02, 21360.071, 21379.122, 21398.173, 21417.224, 21436.275, 21455.326, 21474.377
```

## 2 Model and data loading

Trained models:

```
temp = readRDS("05_final_model.rds")
fit.gg = temp$gg
fit.km0 = temp$km0
data.nswpcn = temp$data.train
```

```
data.glasgow = readRDS("06_Glasgow.rds")
data.glasgow = data.glasgow[data.glasgow$Path.Type %in% c("Pancreatic Adenocarcinoma", "Pancreatic adenocarcinoma"), ]
data.glasgow$Path.LN.Negative = data.glasgow$Path.LN.Inspected - data.glasgow$Path.LN.Involved
data.glasgow$History.Diagnosis.AgeAt = data.glasgow$History.Diagnosis.AgeAt.Cent + 68
data.glasgow$Path.Size = data.glasgow$Path.Size.Cent + 30
data.glasgow$SexM = data.glasgow$Patient.Sex == "M"
data.glasgow$AgeCent = data.glasgow$History.Diagnosis.AgeAt.Cent
data.glasgow$SizeCent = data.glasgow$Path.Size.Cent
data.glasgow$A2 = data.glasgow$Molec.S100A2.DCThresh
data.glasgow$A4 = data.glasgow$Molec.S100A4.DCThresh
data.glasgow$LocBody = data.glasgow$Path.Location != "HOP"
data.glasgow$Time = data.glasgow$History.Death.EventTimeDays
data.glasgow$DSD = data.glasgow$History.DSDeath.Event
```

```
scores.apgi = read.csv("../data/APGI_20150214.csv")
data.apgi = readRDS("../biosurv/data/01_cpvs.rds")
data.apgi$A2 = scores.apgi$A2[match(data.apgi$Patient.ID, scores.apgi$PatientID)]
data.apgi$A4 = scores.apgi$A4[match(data.apgi$Patient.ID, scores.apgi$PatientID)]
rm(scores.apgi)

data.apgi$Path.LN.Inspected = data.apgi$Path.Nodes.Regional.Total
data.apgi$Path.LN.Involved = data.apgi$Path.Nodes.Regional.Involved
data.apgi$Path.LN.Negative = data.apgi$Path.LN.Inspected - data.apgi$Path.LN.Involved
data.apgi$History.Diagnosis.AgeAt = data.apgi$History.Diagnosis.AgeAtYears
data.apgi$History.Diagnosis.AgeAt.Cent = data.apgi$History.Diagnosis.AgeAt - 68
data.apgi$Path.Size = data.apgi$Path.TumourSizeMm
data.apgi$Path.Size.Cent = data.apgi$Path.Size - 30
data.apgi$Patient.Sex = data.apgi$Patient.Gender
data.apgi$SexM = data.apgi$Patient.Sex == "M"
data.apgi$Treat.MarginPositive = data.apgi$Treat.Surgery.ExcisionStatus != "R0"
data.apgi$AgeCent = data.apgi$History.Diagnosis.AgeAt.Cent
data.apgi$SizeCent = data.apgi$Path.Size.Cent
data.apgi$A2 = data.apgi$A2 == 1
data.apgi$A4 = data.apgi$A4 == 1
data.apgi$Stage.pT = data.apgi$Staging.pT
data.apgi$Stage.pT.Simplified = c("T1" = "T1", "T2" = "T2", "T3" = "T34", "T4" = "T34")[as.character(data.apgi$Stage.pT)]
data.apgi$Path.LocationBody = !grepl("head", data.apgi$Path.TumourLocation, ignore.case = TRUE)
data.apgi$Path.LocationBody[data.apgi$Path.TumourLocation == ""] = NA
data.apgi$Path.Differentiation = data.apgi$Path.Grade
data.apgi$LocBody = data.apgi$Path.LocationBody
data.apgi$Time = data.apgi$Surv.EventTimeFromSurg.DSDeath
data.apgi$DSD = data.apgi$Surv.Event.DSDeath
```

```
temp.sel = apply(!is.na(data.apgi[,c("Path.LN.Inspected", "Path.LN.Involved", "Path.LN.Negative", "SexM", "SexF")]), 2, FUN=function(x) {
data.apgi = data.apgi[temp.sel,]
```

```
summary(data.nswpcn)
```

```
##      Patient.ID      Patient.Sex Cohort.ICGC      History.PreviousMalignancy
## Min.      : 4      F:120      Mode :logical      Mode :logical
## 1st Qu.: 305      M:120      FALSE:240      FALSE:219
## Median : 621      NA's :0      TRUE :21
## Mean      : 618      NA's :0
## 3rd Qu.:1030
## Max.      :1453
##
## History.FdrWithPancCancer History.FdrWithAnyCancer History.Diagnosis.Date
## Mode :logical      Mode :logical      Min.      :1994-03-09
## FALSE:230      FALSE:202      1st Qu.:1998-06-26
## TRUE :8      TRUE :38      Median :2001-05-24
## NA's :2      NA's :0      Mean      :2000-12-19
##      3rd Qu.:2003-06-16
##      Max.      :2006-08-14
##
## History.Diagnosis.AgeAt History.AlcoholLevel History.Smoking.Status
## Min.      :28.0      0:151      Never      :140
## 1st Qu.:62.0      1: 45      Ceased : 48
## Median :69.0      2: 22      Current: 52
## Mean      :67.5      3: 22
## 3rd Qu.:75.0
## Max.      :87.0
##
## History.Smoking.PackYears History.Comorbid.Diabetes
## Min.      : 2.0      Mode :logical
## 1st Qu.:20.0      FALSE:181
## Median :25.0      TRUE :59
## Mean      :31.9      NA's :0
## 3rd Qu.:50.0
## Max.      :80.0
## NA's      :185
## History.Comorbid.ChronicPancreatitis History.Recurrence.Event
## Mode :logical      Min.      :0.000
## FALSE:229      1st Qu.:1.000
## TRUE :11      Median :1.000
## NA's :0      Mean      :0.971
##      3rd Qu.:1.000
##      Max.      :1.000
##
## History.Recurrence.Date History.DSDeath.Event History.Death.Date
## Min.      :1994-07-21      Min.      :0.000      Min.      :1995-01-12
## 1st Qu.:1999-09-16      1st Qu.:1.000      1st Qu.:1999-11-30
## Median :2002-06-03      Median :1.000      Median :2002-11-21
## Mean      :2002-03-05      Mean      :0.963      Mean      :2002-08-01
## 3rd Qu.:2005-01-08      3rd Qu.:1.000      3rd Qu.:2005-04-21
## Max.      :2009-01-29      Max.      :1.000      Max.      :2011-10-03
## NA's      :79
```

```

## History.Followup.Date History.Death.EventTimeDays Treat.Resected
## Min. :2009-10-24 Min. : 26 Mode:logical
## 1st Qu.:2009-10-24 1st Qu.: 274 TRUE:240
## Median :2009-10-24 Median : 476 NA's:0
## Mean :2010-01-06 Mean : 592
## 3rd Qu.:2010-02-12 3rd Qu.: 771
## Max. :2010-06-03 Max. :2701
## NA's :237
## Treat.ProcedureWhipple Treat.MarginPositive Treat.Chemo.Any
## Mode :logical Mode :logical Mode :logical
## FALSE:44 FALSE:137 FALSE:97
## TRUE :196 TRUE :103 TRUE :117
## NA's :0 NA's :0 NA's :26
##
##
##
## Treat.Chemo.Adjuvant Treat.Chemo.Adjuvant.GE3Cycles
## Mode :logical Mode :logical
## FALSE:169 FALSE:197
## TRUE :71 TRUE :43
## NA's :0 NA's :0
##
##
##
## Treat.Chemo.Palliative Treat.Chemo.PalliativeDC Treat.Chemo.GEM
## Mode :logical Mode :logical Mode :logical
## FALSE:1 FALSE:170 FALSE:151
## TRUE :65 TRUE :70 TRUE :88
## NA's :174 NA's :0 NA's :1
##
##
##
## Treat.Radio Path.LocationBody Path.Size Path.Bilirubin.Preop
## Mode :logical Mode :logical Min. : 8.0 Min. : 0.06
## FALSE:197 FALSE:196 1st Qu.:25.0 1st Qu.: 0.69
## TRUE :43 TRUE :44 Median :30.0 Median : 3.63
## NA's :0 NA's :0 Mean :33.6 Mean : 7.31
## 3rd Qu.:40.0 3rd Qu.:10.72
## Max. :90.0 Max. :45.03
## NA's :96
##
## Path.Ca199.Preop Path.Bilirubin.Postop Path.Ca199.Postop
## Min. : 1 Min. : 0.12 Min. : 1
## 1st Qu.: 73 1st Qu.: 0.47 1st Qu.: 17
## Median : 218 Median : 0.70 Median : 77
## Mean : 2803 Mean : 1.95 Mean : 1571
## 3rd Qu.: 842 3rd Qu.: 1.30 3rd Qu.: 278
## Max. :101075 Max. :25.38 Max. :31760
## NA's :162 NA's :100 NA's :137
##
## Path.Subtype Path.Differentiation Path.LN.Involved
## Adenosquamous: 18 1: 16 Min. : 0.00
## Large Cell : 0 2:157 1st Qu.: 0.00
## Mucinous : 5 3: 67 Median : 1.00
## NotSpecified : 38 4: 0 Mean : 1.76
## Papillary : 2 3rd Qu.: 2.00

```

```

## Tubular      :177                      Max.      :12.00
##                                         NA's       :3
## Path.LN.Inspected Path.Invasion.Vascular Path.Invasion.Perineural
## Min.      : 0.00      Mode :logical      Mode :logical
## 1st Qu.: 5.00      FALSE:128      FALSE:58
## Median : 8.00      TRUE :112      TRUE :182
## Mean      : 9.68      NA's :0      NA's :0
## 3rd Qu.:13.00
## Max.      :52.00
## NA's      :20
## Stage.pT Stage.pN Stage.pM Molec.BNIP3.NucInt Molec.BNIP3.CytoInt
## Tis: 0 NO : 80 M0 :177 0 : 6 0 : 1
## T1 : 18 N1 :156 M1 : 8 1 :200 1 :125
## T2 : 32 NA's: 4 NA's: 55 2 : 21 2 : 74
## T3 :190 3 : 2 3 : 29
## T4 : 0 NA's: 11 NA's: 11
##
##
## Molec.CCND1.CytoLo Molec.CCND1.CytoHi Molec.CCND1.MembLo
## 0 :152 0 :71 0 :96
## 1 : 34 1 :87 1 :68
## 2 : 4 2 :32 2 :18
## 3 : 1 3 : 1 3 : 9
## NA's: 49 NA's:49 NA's:49
##
##
## Molec.CCND1.MembHi Molec.Grb7.Int Molec.Grb7.Percent Molec.HCNT3PlusHENT1
## 0 :29 0 :49 Min. : 0.0 Mode :logical
## 1 :86 1 :90 1st Qu.: 3.0 FALSE:93
## 2 :45 2 :42 Median : 18.0 TRUE :94
## 3 :31 3 : 7 Mean : 31.6 NA's :53
## NA's:49 NA's:52 3rd Qu.: 58.5
## Max. :100.0
## NA's :52
##
## Molec.HENT1.Percent Molec.HENT1.Int Molec.HER2 Molec.HOXB2.Percent
## Min. : 0.0 0 : 17 Mode :logical Min. : 0.0
## 1st Qu.: 11.2 1 :114 FALSE:36 1st Qu.: 35.0
## Median : 42.5 2 : 51 TRUE :10 Median : 70.0
## Mean : 44.4 3 : 12 NA's :194 Mean : 59.6
## 3rd Qu.: 75.0 NA's: 46 3rd Qu.: 85.0
## Max. :100.0 Max. :100.0
## NA's :46 NA's :42
##
## Molec.HOXB2.Int Molec.RON.Int Molec.S100A2.Int Molec.S100A2.Percent
## 0 : 14 0 : 19 0:87 Min. : 0.0
## 1 :137 1 :110 1:59 1st Qu.: 0.0
## 2 : 33 2 : 59 2:56 Median : 10.0
## 3 : 14 3 : 10 3:38 Mean : 28.1
## NA's: 42 NA's: 42 3rd Qu.: 60.0
## Max. :100.0
##
## Molec.S100A2.StromaScore Molec.S100A4.CytoInt Molec.S100A4.CytoPercent
## Mode :logical 0:70 Min. : 0.0
## FALSE:175 1:89 1st Qu.: 0.0
## TRUE :22 2:40 Median : 10.0

```

```

## NA's :43          3:41          Mean   : 34.8
##                                     3rd Qu.: 75.0
##                                     Max.   :100.0
##
## Molec.S100A4.NucInt Molec.S100A4.NucPercent Stage.Overall
## 0:78                Min.   : 0.0          IIB    :117
## 1:66                1st Qu.: 0.0          IIA    : 41
## 2:62                Median : 5.0          IB     : 12
## 3:34                Mean    : 26.4         IV     : 8
##                   3rd Qu.: 60.0         IA     : 7
##                   Max.     :100.0        (Other): 0
##                                     NA's    : 55
## History.Death.Event Molec.S100A4.DCThresh Molec.S100A2.DCThresh
## Min.   :0.000      Mode :logical      Mode :logical
## 1st Qu.:1.000      FALSE:60          FALSE:203
## Median :1.000      TRUE :180          TRUE :37
## Mean    :0.996      NA's :0           NA's :0
## 3rd Qu.:1.000
## Max.     :1.000
##
## Stage.pT.Simplified Path.Ca199.Preop.Cent Path.Ca199.Postop.Cent
## T1 : 18            Min.   : -5.38        Min.   : -3.97
## T2 : 32            1st Qu.: -1.09        1st Qu.: -1.14
## T34:190           Median : 0.00          Median : 0.37
##                   Mean    : 0.09          Mean    : 0.62
##                   3rd Qu.: 1.36          3rd Qu.: 1.66
##                   Max.     : 6.14          Max.     : 6.40
##                   NA's     :162          NA's     :137
## History.Diagnosis.AgeAt.Cent History.Smoking.PackYears.Cent
## Min.   : -40.00      Min.   : -28.00
## 1st Qu.: -6.00       1st Qu.: -10.00
## Median : 1.00        Median : -5.00
## Mean    : -0.51       Mean    : 1.89
## 3rd Qu.: 7.00        3rd Qu.: 20.00
## Max.     : 19.00      Max.     : 50.00
##                                     NA's     :185
## Path.Size.Cent      Path.Bilirubin.Preop.Cent Path.Bilirubin.Postop.Cent
## Min.   : -22.00      Min.   : -3.39        Min.   : -0.53
## 1st Qu.: -5.00       1st Qu.: -2.76        1st Qu.: -0.18
## Median : 0.00        Median : 0.18          Median : 0.06
## Mean    : 3.56       Mean    : 3.86          Mean    : 1.30
## 3rd Qu.: 10.00       3rd Qu.: 7.27          3rd Qu.: 0.66
## Max.     : 60.00      Max.     :41.58          Max.     :24.74
##                                     NA's     :96          NA's     :100
## History.Diagnosis.Date.Cent Path.LN.InvolvedFraction Path.LN.Negative
## Min.   : -2867       Min.   : 0.000         Min.   : 0.00
## 1st Qu.: -1297       1st Qu.: 0.000         1st Qu.: 4.00
## Median : -234        Median : 0.143          Median : 7.00
## Mean    : -389       Mean    : 0.217          Mean    : 7.85
## 3rd Qu.: 519         3rd Qu.: 0.333          3rd Qu.:11.00
## Max.     : 1674       Max.     :1.000          Max.     :45.00
##                                     NA's     :21          NA's     :20
## SexM                Ca199                DiagYearCent                Time
## Mode :logical      Mode :logical      Min.   : -7.849      Min.   : 26

```



```

## FALSE:120      FALSE:26      1st Qu.: -3.551  1st Qu.: 274
## TRUE :120      TRUE :52       Median : -0.639  Median : 474
## NA's :0       NA's :162      Mean  : -1.065  Mean  : 589
##                                     3rd Qu.: 1.422  3rd Qu.: 764
##                                     Max.   : 4.583  Max.   :2701
##
##      DSD      AgeCent      LocBody      SizeCent
## Mode :logical Min.   : -40.00 Mode :logical Min.   : -22.00
## FALSE:9      1st Qu.: -6.00 FALSE:196      1st Qu.: -5.00
## TRUE :231     Median :  1.00 TRUE :44       Median :  0.00
## NA's :0      Mean  : -0.51 NA's :0        Mean  :  3.56
##              3rd Qu.:  7.00      3rd Qu.: 10.00
##              Max.   : 19.00      Max.   : 60.00
##
##      A2      A4      SizePlus
## Mode :logical Mode :logical Min.   : 0.00
## FALSE:203     FALSE:60      1st Qu.: 0.00
## TRUE :37      TRUE :180      Median : 0.00
## NA's :0       NA's :0        Mean  : 7.35
##              3rd Qu.:10.00
##              Max.   :60.00
##
summary(data.glasgow)

## Patient.ID      Patient.Sex History.Diagnosis.AgeAt Treat.Procedure
## Length:189      F: 89      Min.   :37.5      Length:189
## Class :character M:100     1st Qu.:57.8      Class :character
## Mode  :character      Median :64.0      Mode  :character
##              Mean  :62.6
##              3rd Qu.:69.4
##              Max.   :86.0
##
## Path.Location
## Length:189
## Class :character
## Mode  :character
##
##
##
## Path.Type
## Pancreatic Adenocarcinoma      :156
## Pancreatic adenocarcinoma      : 32
## Pancreatic Adenocarcinoma      :  1
## Pancreatic adenocarcinoma arising form IPMN      :  0
## Pancreatic adenocarcinoma arising from mucinous cystic neoplasm:  0
## Pancreatic Adenocarcinoma arising IPMN      :  0
## (Other)      :  0
## Path.Differentiation Path.Grade Stage.pT Stage.pN
## 1: 12      Low :128 Tis:  0 N0: 33
## 2:117      High: 61 T1 :  1 N1:156
## 3: 60      T2 : 13
## 4:  0      T3 :171

```

```

##                                T4 : 4
##
##
## Path.Invasion.Perineural Path.Invasion.Vascular Path.LN.Inspected
## Mode :logical           Mode :logical           Min.    : 1.0
## FALSE:13                FALSE:96                1st Qu.:13.0
## TRUE :176                TRUE :93                 Median :20.0
## NA's :0                  NA's :0                 Mean    :20.2
##                               3rd Qu.:27.0
##                               Max.    :53.0
##
## Path.LN.InvolvedFraction Treat.MarginPositive Treat.VeinResection
## Min.    :0.00           Mode :logical           Mode :logical
## 1st Qu.:0.05           FALSE:51           FALSE:158
## Median :0.14           TRUE :138           TRUE :31
## Mean    :0.20           NA's :0             NA's :0
## 3rd Qu.:0.27
## Max.    :1.00
##
## Path.Size History.Death.EventTimeDays History.Death.Cause
## Min.    : 5.0 Min.    : 8 0: 9
## 1st Qu.:25.0 1st Qu.: 233 1:161
## Median :30.0 Median : 501 2: 19
## Mean    :32.7 Mean    : 673
## 3rd Qu.:40.0 3rd Qu.: 915
## Max.    :65.0 Max.    :3531
##
## Treat.Chemo.Adjuvant Treat.Chemo.Neoadjuvant Molec.S100A2.DCThresh
## Mode :logical           Mode :logical           Mode :logical
## FALSE:110              FALSE:188              FALSE:127
## TRUE :79               TRUE :1                TRUE :62
## NA's :0                NA's :0                NA's :0
##
##
##
## Molec.S100A4.DCThresh Treat.ProcedureWhipple Path.LocationBody
## Mode :logical           Mode:logical           Mode :logical
## FALSE:55              TRUE:189              FALSE:189
## TRUE :134             NA's:0                NA's :0
## NA's :0
##
##
##
## History.DSDeath.Event History.ACDeath.Event Path.LN.Involved
## Mode :logical           Mode :logical           Min.    : 0.00
## FALSE:28              FALSE:9                1st Qu.: 1.00
## TRUE :161             TRUE :180              Median : 2.00
## NA's :0               NA's :0                Mean    : 3.57
##                               3rd Qu.: 5.00
##                               Max.    :32.00
##
##
## History.Diagnosis.AgeAt.Cent Path.Size.Cent Stage.pT.Simplified
## Min.    : -30.55 Min.    : -25.00 T1 : 1
## 1st Qu.: -10.19 1st Qu.: -5.00 T2 : 13

```

```

## Median : -4.00          Median : 0.00   T34:175
## Mean   : -5.37          Mean    : 2.72
## 3rd Qu.: 1.43           3rd Qu.: 10.00
## Max.   : 18.00          Max.    : 35.00
##
## Path.LN.Negative      SexM          AgeCent          SizeCent
## Min.    : 0.0         Mode :logical  Min.    :-30.55   Min.    :-25.00
## 1st Qu.:10.0         FALSE:89   1st Qu.: -10.19  1st Qu.: -5.00
## Median :16.0         TRUE :100   Median : -4.00   Median : 0.00
## Mean    :16.6        NA's :0        Mean    : -5.37   Mean    : 2.72
## 3rd Qu.:23.0         3rd Qu.: 1.43  3rd Qu.: 10.00
## Max.    :47.0        Max.    : 18.00  Max.    : 35.00
##
##      A2              A4              LocBody              Time
## Mode :logical      Mode :logical      Mode :logical      Min.    : 8
## FALSE:127          FALSE:55          FALSE:189          1st Qu.: 233
## TRUE :62           TRUE :134          NA's :0            Median : 501
## NA's :0            NA's :0            Mean    : 673
##                                     3rd Qu.: 915
##                                     Max.    :3531
##
##      DSD
## Mode :logical
## FALSE:28
## TRUE :161
## NA's :0
##
##
##
summary(data.apgi)

## Patient.ID          Patient.Gender          Patient.Ethnicity
## Length:75          Female:34          Asian          : 7
## Class :character    Male :41          Asian, White/Caucasian : 0
## Mode :character          Black/African          : 1
##                                     Black/African, White/Caucasian: 0
##                                     White/Caucasian          :67
##
##
##      Patient.Country History.LastFollowup.Date
## Australia          :75          Min.    :2008-04-14
## Italy              : 0          1st Qu.:2011-02-03
## New Zealand        : 0          Median :2012-05-09
## Puerto Rico        : 0          Mean    :2012-06-02
## United Kingdom     : 0          3rd Qu.:2013-11-06
## United States of America: 0      Max.    :2014-09-08
##
## History.Smoking.PackYears History.Diagnosis.Date
## Min.    : 0.75          Min.    :2004-12-30
## 1st Qu.: 12.00          1st Qu.:2009-11-28
## Median : 27.50          Median :2010-05-28
## Mean    : 30.98          Mean    :2010-06-08
## 3rd Qu.: 44.06          3rd Qu.:2010-11-29

```

```

## Max. :123.75 Max. :2012-02-17
## NA's :43
## History.Diagnosis.AgeAtYears History.Surgery.Date
## Min. :47.0 Min. :2004-12-30
## 1st Qu.:60.5 1st Qu.:2009-12-05
## Median :67.0 Median :2010-06-01
## Mean :66.8 Mean :2010-06-16
## 3rd Qu.:74.0 3rd Qu.:2011-01-19
## Max. :84.0 Max. :2012-02-17
##
##
## Treat.Surgery.Procedure
## Classic Whipple :55
## Classic Whipple, Exploratory laparotomy : 3
## PPPD : 3
## Splenectomy, Subtotal Panc/L sided Panc or distal Panc : 3
## Subtotal Panc/L sided Panc or distal Panc : 3
## Cholecystectomy, Cholecystojejunostomy/Hepaticojejunostomy, Classic Whipple: 1
## (Other) : 7
## Treat.Surgery.ExcisionStatus Treat.Surgery.Margin.Pancreatic
## R0:51 <2 mm : 2
## R1:20 Clear :65
## R2: 4 Involved: 2
## NA's : 6
##
##
##
## Treat.Surgery.MarginSizeMm.Pancreatic Treat.Surgery.Margin.Periunc
## Min. : 0.00 <2 mm :16
## 1st Qu.: 5.00 Clear :36
## Median :10.00 Involved:11
## Mean : 9.94 NA's :12
## 3rd Qu.:10.00
## Max. :40.00
## NA's :15
## Treat.Surgery.MarginSizeMm.Periunc Treat.Surgery.Margin.PVGroove
## Min. : 0.00 <2 mm :18
## 1st Qu.: 1.00 Clear :37
## Median : 2.20 Involved:10
## Mean : 6.92 NA's :10
## 3rd Qu.:10.00
## Max. :40.00
## NA's :24
## Treat.Surgery.MarginSizeMm.PVGroove Treat.Surgery.Margin.Retrop
## Min. : 0.0 <2 mm :19
## 1st Qu.: 1.0 Clear :46
## Median : 2.0 Involved: 5
## Mean : 3.8 NA's : 5
## 3rd Qu.: 4.0
## Max. :25.0
## NA's :24
## Treat.Surgery.MarginSizeMm.Retrop Treat.Surgery.Margin.CBD
## Min. : 0.10 <2 mm : 0
## 1st Qu.: 1.00 Clear :58
## Median : 3.00 Involved: 0

```

```

## Mean      : 5.29                      NA's      :17
## 3rd Qu.: 8.00
## Max.      :25.00
## NA's      :14
## Treat.Surgery.MarginSizeMm.CBD Treat.Surgery.Margin.Duodenal
## Min.      : 3.0                      Clear      :40
## 1st Qu.:11.5                      Involved: 0
## Median :20.0                      NA's      :35
## Mean      :21.9
## 3rd Qu.:30.0
## Max.      :50.0
## NA's      :31
## Treat.Surgery.MarginSizeMm.Duodenal Treat.Surgery.Margin.Gastric
## Min.      : 20.0                      Clear:39
## 1st Qu.: 47.5                      NA's :36
## Median : 75.0
## Mean      : 75.0
## 3rd Qu.:102.5
## Max.      :130.0
## NA's      :73
## Treat.Surgery.MarginSizeMm.Gastric Treat.Surgery.Margin.Comments
## Min.      : 20                      Length:75
## 1st Qu.: 40                      Class :character
## Median : 60                      Mode  :character
## Mean      : 60
## 3rd Qu.: 80
## Max.      :100
## NA's      :73
##
##                      Path.HistoType
## Pancreatic Ductal Adenocarcinoma:75
## Acinar Cell Carcinoma             : 0
## Ampullary Adenocarcinoma          : 0
## Carcinoid Tumour                  : 0
## Cholangiocarcinoma                : 0
## Clear Cell Carcinoma               : 0
## (Other)                           : 0
##
##                      Path.HistoType.Subtype Path.Grade
## Gastric                  : 0                1: 3
## Intestinal               : 0                2:47
## Mixed                    : 0                3:23
## Not otherwise Specified (NOS):10           4: 2
## Pancreatobiliary         :10
## Squamous                 : 0
## NA's                     :55
##
## Path.TumourLocation Path.TumourSizeMm Path.Invasion.PN
## Head                 :55      Min.    :15.0      Absent : 9
## Head (Uncinate): 9      1st Qu.:28.0      Present:66
## Body                 : 7      Median :35.0
## Tail                 : 3      Mean    :36.9
## Ampulla              : 1      3rd Qu.:43.0
##                      : 0      Max.    :90.0
## (Other)              : 0
## Path.Invasion.VS Path.Nodes.Regional.Total Path.Nodes.Regional.Involved
## Absent :22      Min.    : 2.0      Min.    : 0.00

```

```

## Present:51      1st Qu.:13.0      1st Qu.: 1.00
## NA's      : 2      Median :16.0      Median : 3.00
##           Mean  :18.6      Mean  : 3.03
##           3rd Qu.:23.5      3rd Qu.: 4.00
##           Max.   :46.0      Max.   :13.00
##
## Path.Nodes.SepRec.Total Path.Nodes.SepRec.Involved
## Min.      : 2.0      Min.      : 0.00
## 1st Qu.:13.0      1st Qu.: 1.00
## Median :16.0      Median : 3.00
## Mean  :18.6      Mean  : 3.03
## 3rd Qu.:23.5      3rd Qu.: 4.00
## Max.   :46.0      Max.   :13.00
##
##
##                               Staging.Version Staging.pM Staging.pN
## pTNM AJCC 6th Ed 2002                               :12      M0 : 2      N0:16
## pTNM AJCC 7th Ed 2010                               :63      M1 : 4      N1:59
## pTNM AJCC 7th Ed 2010 (Ampulla)                     : 0      NA's:69
## pTNM AJCC 7th Ed 2010 (Cholangiocarcinoma): 0
## pTNM AJCC 7th Ed 2010 (Neuroendocrine)              : 0
##
##
## Staging.pT Staging.Stage History.Recurrence History.Recurrence.Date
## Tis: 0      IA : 1      Not observed:15      Min.      :2007-12-31
## T1 : 1      IB : 1      Suspected : 2      1st Qu.:2010-10-25
## T2 : 3      IIA:13      Confirmed :56      Median :2011-04-11
## T3 :70      IIB:55      NA's : 2      Mean :2011-06-29
## T4 : 1      III: 1      3rd Qu.:2012-02-28
##           IV : 4      Max. :2014-08-27
##                               NA's :17
##
## History.Recurrence.Site.Stomach History.Recurrence.Site.Peritoneum
## Mode :logical      Mode :logical
## FALSE:75      FALSE:67
## NA's :0      TRUE :8
##           NA's :0
##
##
## History.Recurrence.Site.PancRemnant History.Recurrence.Site.PancBed
## Mode :logical      Mode :logical
## FALSE:70      FALSE:64
## TRUE :5      TRUE :11
## NA's :0      NA's :0
##
##
## History.Recurrence.Site.Other History.Recurrence.Site.Omentum
## Mode :logical      Mode :logical
## FALSE:69      FALSE:74
## TRUE :6      TRUE :1
## NA's :0      NA's :0
##
##
##

```

```

## History.Recurrence.Site.Mesentery History.Recurrence.Site.LymphNodes
## Mode :logical Mode :logical
## FALSE:74 FALSE:61
## TRUE :1 TRUE :14
## NA's :0 NA's :0
##
##
##
## History.Recurrence.Site.Lung History.Recurrence.Site.Liver
## Mode :logical Mode :logical
## FALSE:60 FALSE:51
## TRUE :15 TRUE :24
## NA's :0 NA's :0
##
##
##
## History.Recurrence.Site.Brain History.Recurrence.Site.Bone
## Mode :logical Mode :logical
## FALSE:73 FALSE:71
## TRUE :2 TRUE :4
## NA's :0 NA's :0
##
##
##
## History.Status History.Death.Date
## Alive - With Disease : 7 Min. :2008-05-13
## Alive - Without Disease :13 1st Qu.:2010-12-20
## Deceased - Of Disease :51 Median :2011-12-28
## Deceased - Of Other Cause : 4 Mean :2011-11-08
## Deceased - Of Unknown Cause: 0 3rd Qu.:2012-09-08
## Max. :2014-01-26
## NA's :20
##
## History.Death.Cause Surv.Event.Death
## Cancer Death (Pancreatic) :51 Min. :0.000
## Died of Treatment Complication : 2 1st Qu.:0.000
## Cancer Death (Other) - Lung ca : 1 Median :1.000
## Other (please specify) - Suicide: 1 Mean :0.733
## Other (please specify) : 0 3rd Qu.:1.000
## (Other) : 0 Max. :1.000
## NA's :20
## Surv.EventTimeFromDiag.Death Surv.EventTimeFromSurg.Death
## Min. : 56 Min. : 62
## 1st Qu.: 386 1st Qu.: 362
## Median : 653 Median : 655
## Mean : 753 Mean : 745
## 3rd Qu.:1007 3rd Qu.:1010
## Max. :2848 Max. :2848
##
## Surv.EventTimeFromRec.Death Surv.Event.DSDeath
## Min. : 3.0 Min. :0.00
## 1st Qu.: 65.8 1st Qu.:0.00
## Median : 202.0 Median :1.00
## Mean : 287.4 Mean :0.68
## 3rd Qu.: 371.2 3rd Qu.:1.00

```

```

## Max. :1333.0 Max. :1.00
## NA's :17
## Surv.EventTimeFromDiag.DSDeath Surv.EventTimeFromSurg.DSDeath
## Min. : 31 Min. : 37
## 1st Qu.: 386 1st Qu.: 362
## Median : 653 Median : 655
## Mean : 752 Mean : 743
## 3rd Qu.:1007 3rd Qu.:1010
## Max. :2848 Max. :2848
##
## Surv.EventTimeFromRec.DSDeath Surv.Event.Recurrence
## Min. : 3.0 Min. :0.000
## 1st Qu.: 65.8 1st Qu.:1.000
## Median : 202.0 Median :1.000
## Mean : 287.1 Mean :0.767
## 3rd Qu.: 371.2 3rd Qu.:1.000
## Max. :1333.0 Max. :1.000
## NA's :17 NA's :2
## Surv.EventTimeFromDiag.Recurrence Surv.EventTimeFromSurg.Recurrence
## Min. : 31 Min. : -15
## 1st Qu.: 241 1st Qu.: 231
## Median : 388 Median : 377
## Mean : 540 Mean : 532
## 3rd Qu.: 698 3rd Qu.: 705
## Max. :1954 Max. :1954
## NA's :2 NA's :2
## A2 A4 Path.LN.Inspected Path.LN.Involved
## Mode :logical Mode :logical Min. : 2.0 Min. : 0.00
## FALSE:64 FALSE:26 1st Qu.:13.0 1st Qu.: 1.00
## TRUE :11 TRUE :49 Median :16.0 Median : 3.00
## NA's :0 NA's :0 Mean :18.6 Mean : 3.03
## 3rd Qu.:23.5 3rd Qu.: 4.00
## Max. :46.0 Max. :13.00
##
## Path.LN.Negative History.Diagnosis.AgeAt History.Diagnosis.AgeAt.Cent
## Min. : 2.0 Min. :47.0 Min. : -21.00
## 1st Qu.: 9.0 1st Qu.:60.5 1st Qu.: -7.50
## Median :13.0 Median :67.0 Median : -1.00
## Mean :15.6 Mean :66.8 Mean : -1.15
## 3rd Qu.:21.0 3rd Qu.:74.0 3rd Qu.: 6.00
## Max. :44.0 Max. :84.0 Max. : 16.00
##
## Path.Size Path.Size.Cent Patient.Sex SexM
## Min. :15.0 Min. : -15.00 Female:34 Mode :logical
## 1st Qu.:28.0 1st Qu.: -2.00 Male :41 FALSE:75
## Median :35.0 Median : 5.00 NA's :0
## Mean :36.9 Mean : 6.89
## 3rd Qu.:43.0 3rd Qu.: 13.00
## Max. :90.0 Max. : 60.00
##
## Treat.MarginPositive AgeCent SizeCent Stage.pT
## Mode :logical Min. : -21.00 Min. : -15.00 Tis: 0
## FALSE:51 1st Qu.: -7.50 1st Qu.: -2.00 T1 : 1
## TRUE :24 Median : -1.00 Median : 5.00 T2 : 3

```



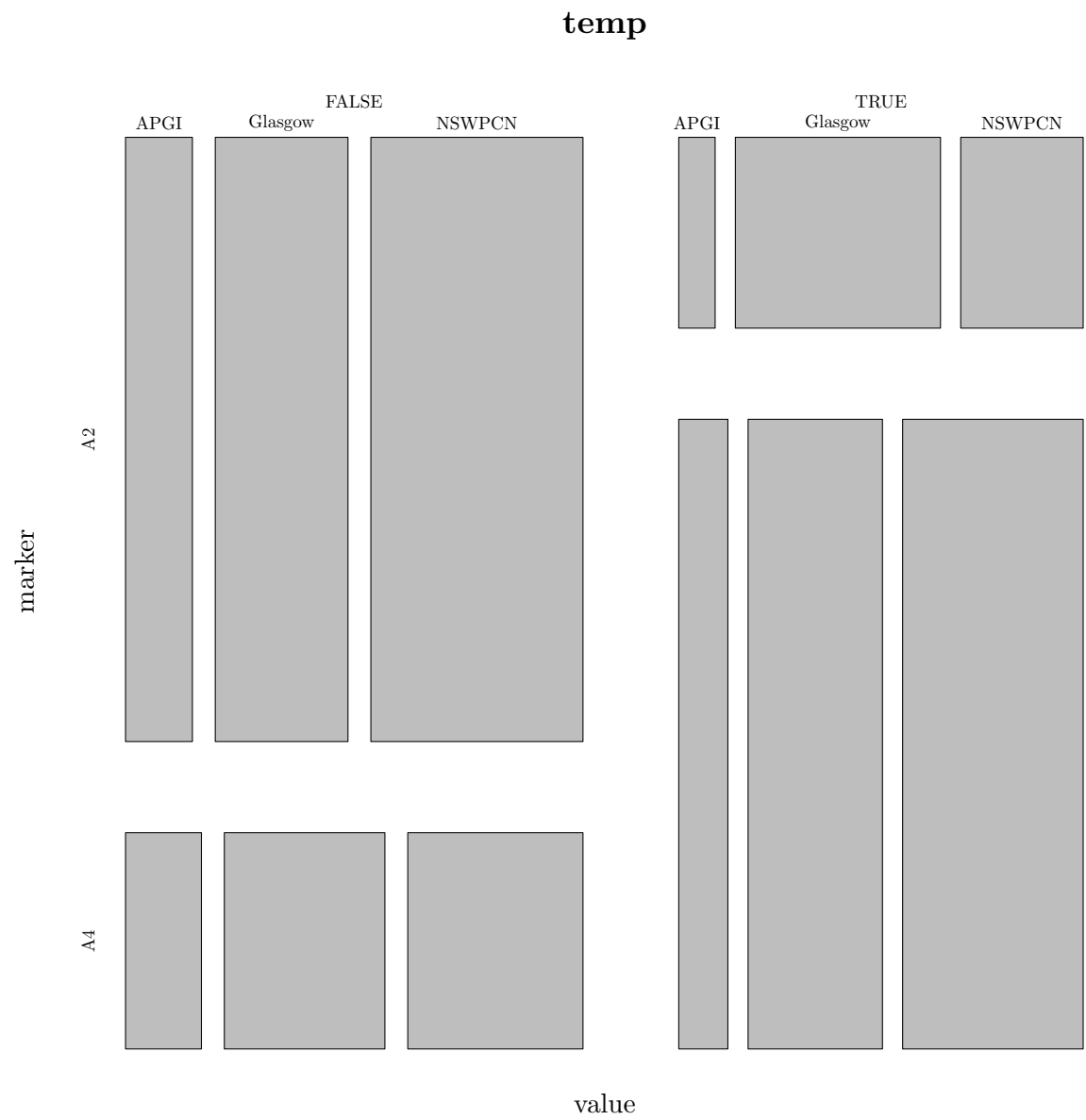
```

## NA's :0          Mean   : -1.15   Mean   :  6.89   T3 :70
##                3rd Qu.:  6.00   3rd Qu.: 13.00   T4 : 1
##                Max.    : 16.00   Max.    : 60.00
##
## Stage.pT.Simplified Path.LocationBody Path.Differentiation
## Length:75          Mode :logical   1: 3
## Class :character   FALSE:64        2:47
## Mode :character    TRUE :11         3:23
##                   NA's :0           4: 2
##
##
##
## LocBody           Time           DSD
## Mode :logical   Min.    :  37     Min.    :0.00
## FALSE:64        1st Qu.: 362     1st Qu.:0.00
## TRUE :11        Median : 655     Median :1.00
## NA's :0         Mean    : 743     Mean    :0.68
##                3rd Qu.:1010     3rd Qu.:1.00
##                Max.    :2848     Max.    :1.00
##
temp = table(value = c(data.nswpcn$A2, data.glasgow$A2, data.apgi$A2, data.nswpcn$A4, data.glasgow$A4, c
temp

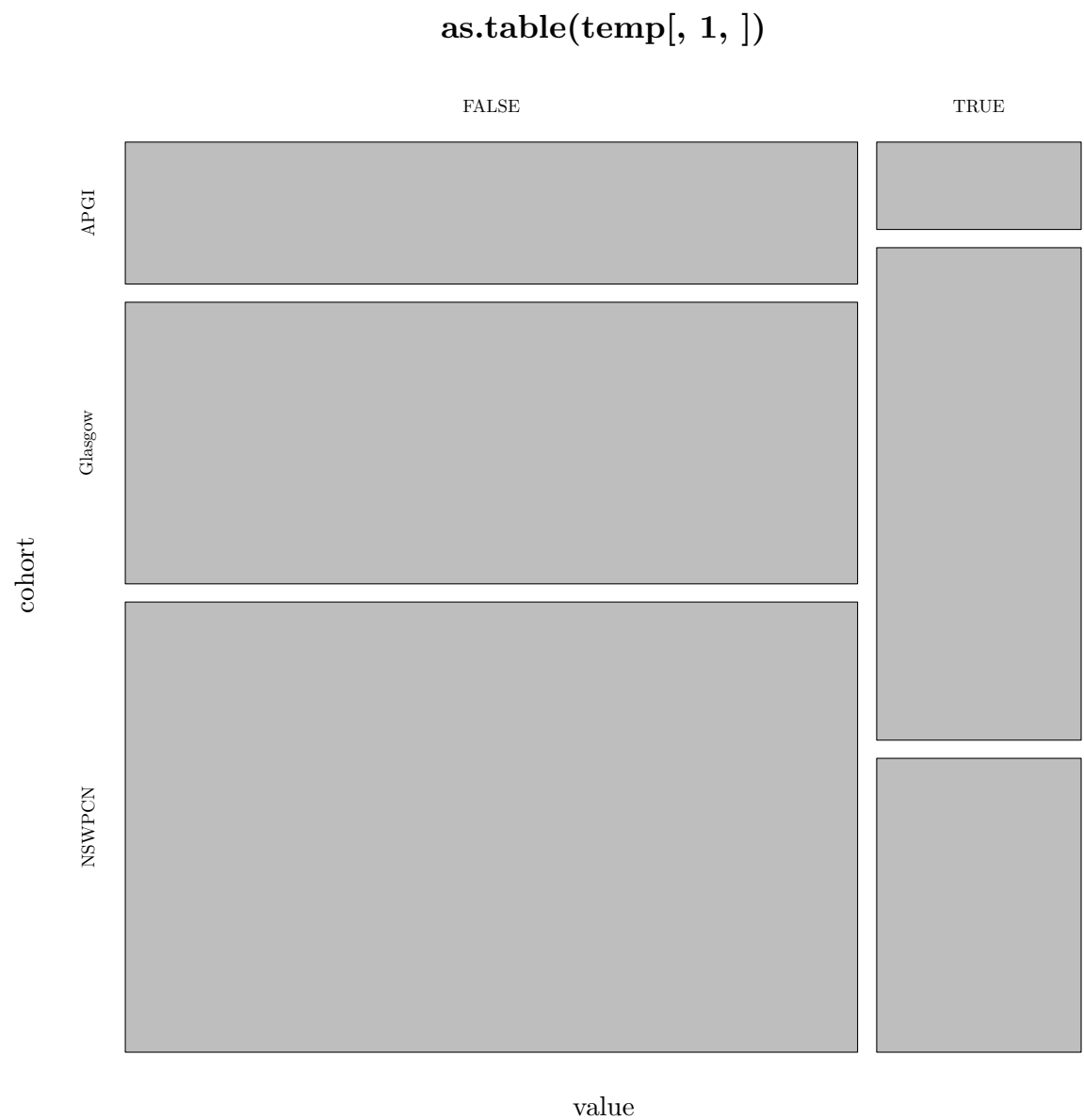
## , , cohort = APGI
##
##      marker
## value    A2  A4
## FALSE   64  26
## TRUE    11  49
##
## , , cohort = Glasgow
##
##      marker
## value    A2  A4
## FALSE  127  55
## TRUE    62 134
##
## , , cohort = NSWPCN
##
##      marker
## value    A2  A4
## FALSE  203  60
## TRUE    37 180

plot(temp)

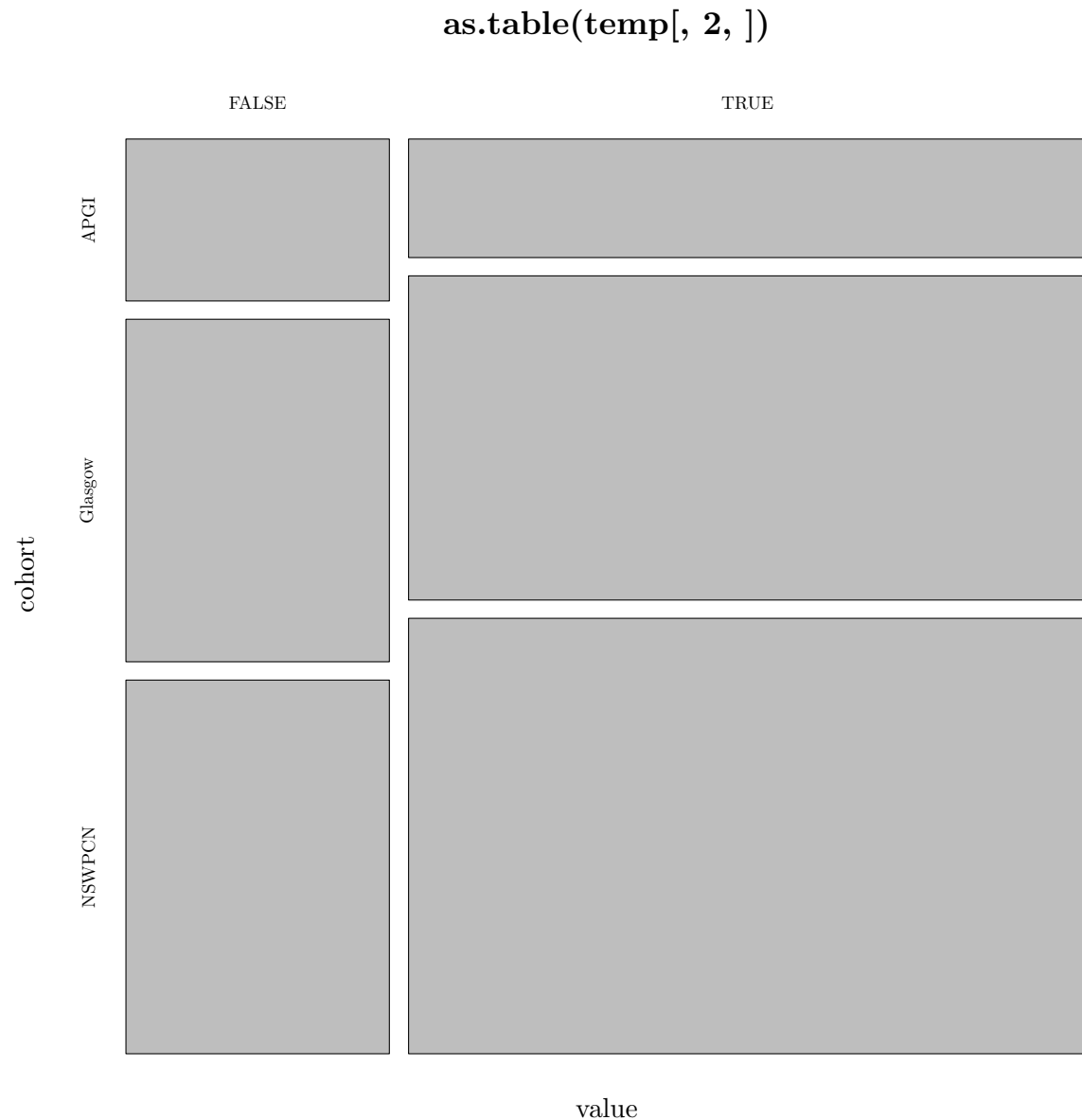
```



```
plot(as.table(temp[,1,]))
```

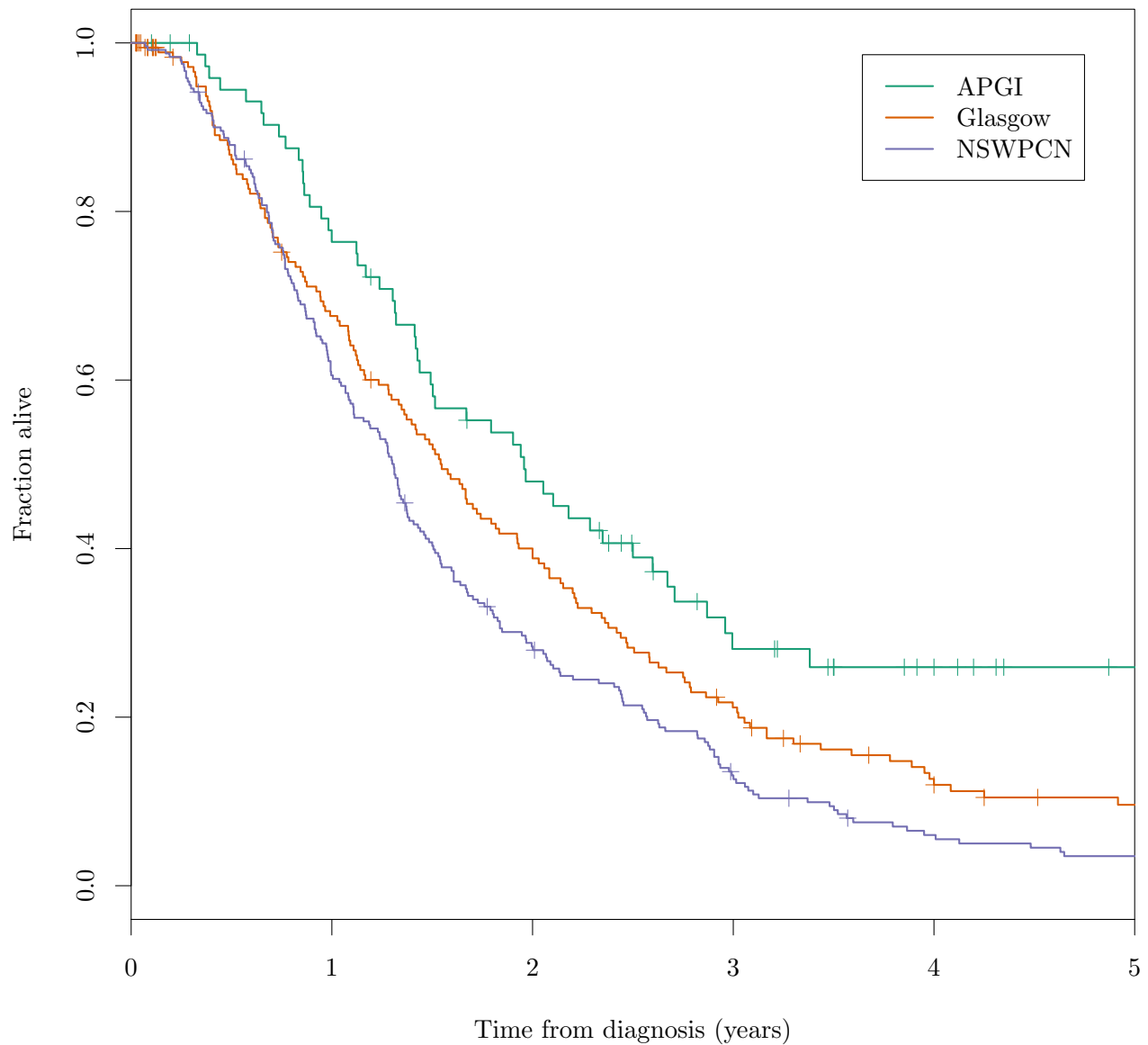


```
plot(as.table(temp[,2,]))
```



```
temp.time = c(data.nswpcn$Time, data.glasgow$Time, data.apgi$Time) / 365.25
temp.dsd = c(data.nswpcn$DSD, data.glasgow$DSD, data.apgi$DSD)
temp.cohort = factor(rep(c("NSWPCN", "Glasgow", "APGI"), c(nrow(data.nswpcn), nrow(data.glasgow), nrow(data.apgi))))
temp.survfit = survfit(Surv(temp.time, temp.dsd) ~ temp.cohort)
plot(temp.survfit, col = pal[1:3], xlim = c(0, 5), lwd = 2, main = "Cohort marginal survival", xlab = "Time (years)")
legend("topright", legend = c("APGI", "Glasgow", "NSWPCN"), col = pal[1:3], inset = 0.05, lwd = 2)
```

## Cohort marginal survival



### 3 Score calculation

```
temp = applyNomogram(fit.mskcc, data.glasgow)

## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Portal.Vein
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Splenectomy
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Posterior.Margin
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Back.pain
```

```

## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Weight.loss

mskcc_post.linpred.glasgow = temp[,1]
mskcc_post.12mo.glasgow = temp[,2]
mskcc_post.24mo.glasgow = temp[,3]
mskcc_post.36mo.glasgow = temp[,4]
temp = applyNomogram(fit.mskcc, data.glasgow[,c("History.Diagnosis.AgeAt", "Patient.Sex", "Path.LocationBo

## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Portal.Vein
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Splenectomy
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Treat.MarginPositive
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Path.Differentiation
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Posterior.Margin
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Path.LN.Involved
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Path.LN.Negative
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Back.pain
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Weight.loss

mskcc_pre.linpred.glasgow = temp[,1]
mskcc_pre.12mo.glasgow = temp[,2]
mskcc_pre.24mo.glasgow = temp[,3]
mskcc_pre.36mo.glasgow = temp[,4]

temp = applyNomogram(fit.mskcc, data.apgi)

## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Portal.Vein
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Splenectomy
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Posterior.Margin
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Back.pain
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Weight.loss

mskcc_post.linpred.apgi = temp[,1]
mskcc_post.12mo.apgi = temp[,2]
mskcc_post.24mo.apgi = temp[,3]
mskcc_post.36mo.apgi = temp[,4]
temp = applyNomogram(fit.mskcc, data.apgi[,c("History.Diagnosis.AgeAt", "Patient.Sex", "Path.LocationBo

## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Portal.Vein
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Splenectomy

```

```
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Treat.MarginPositive
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Path.Differentiation
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Posterior.Margin
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Path.LN.Involved
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Path.LN.Negative
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Back.pain
## Warning in FUN(c("History.Diagnosis.AgeAt", "Patient.Sex", "Portal.Vein", : Marginalizing
missing variable: Weight.loss

mskcc_pre.linpred.apgi = temp[,1]
mskcc_pre.12mo.apgi = temp[,2]
mskcc_pre.24mo.apgi = temp[,3]
mskcc_pre.36mo.apgi = temp[,4]
```

Get approximate linear predictors from the GG model, by just calculating the location term.

```
val.prob.times = seq(0, max(c(data.glasgow$Time, data.apgi$Time)), 1)
```

```
gg.path.glasgow = summary(fit.gg, newdata = data.glasgow, ci = FALSE)
temp.coefs = coef(fit.gg)
gg.linpred.glasgow = sapply(1:length(temp.coefs), function(coef_i) {
  # if (names(temp.coefs)[coef_i] == "SexMTRUE") {
  #   rep(0, nrow(data.val))
  # } else
  if (names(temp.coefs)[coef_i] %in% colnames(data.glasgow)) {
    temp.coefs[coef_i] * data.glasgow[,names(temp.coefs)[coef_i]]
  } else if (gsub("TRUE$", "", names(temp.coefs)[coef_i]) %in% colnames(data.glasgow)) {
    temp.coefs[coef_i] * data.glasgow[,gsub("TRUE$", "", names(temp.coefs)[coef_i])]
  } else {
    rep(0, nrow(data.glasgow))
  } })
gg.linpred.glasgow = -rowSums(gg.linpred.glasgow) # Negate to bring into concordance with the dir
temp = summary(fit.gg, newdata = data.glasgow, ci = FALSE)
gg.prob.glasgow = sapply(temp, function(x) approx(x[,1], x[,2], xout = val.prob.times, yleft = 1, yright = 0))
colnames(gg.prob.glasgow) = rownames(data.glasgow)
```

```
gg.path.apgi = summary(fit.gg, newdata = data.apgi, ci = FALSE)
temp.coefs = coef(fit.gg)
gg.linpred.apgi = sapply(1:length(temp.coefs), function(coef_i) {
  # if (names(temp.coefs)[coef_i] == "SexMTRUE") {
  #   rep(0, nrow(data.val))
  # } else
  if (names(temp.coefs)[coef_i] %in% colnames(data.apgi)) {
    temp.coefs[coef_i] * data.apgi[,names(temp.coefs)[coef_i]]
  } else if (gsub("TRUE$", "", names(temp.coefs)[coef_i]) %in% colnames(data.apgi)) {
    temp.coefs[coef_i] * data.apgi[,gsub("TRUE$", "", names(temp.coefs)[coef_i])]
  } else {
    rep(0, nrow(data.apgi))
  } })
```

```

    } else {
      rep(0, nrow(data.apgi))
    } })
gg.linpred.apgi = -rowSums(gg.linpred.apgi) # Negate to bring into concordance with the direction of
temp = summary(fit.gg, newdata = data.apgi, ci = FALSE)
gg.prob.apgi = sapply(temp, function(x) approx(x[,1], x[,2], xout = val.prob.times, yleft = 1, yright = 1))
colnames(gg.prob.apgi) = rownames(data.apgi)

gg.linpred.nswpcn = sapply(1:length(temp.coefs), function(coef_i) {
  # if (names(temp.coefs)[coef_i] == "SexMTRUE") {
  #   rep(0, nrow(data.val))
  # } else
  if (names(temp.coefs)[coef_i] %in% colnames(data.glasgow)) {
    temp.coefs[coef_i] * data.nswpcn[,names(temp.coefs)[coef_i]]
  } else if (gsub("TRUE$", "", names(temp.coefs)[coef_i]) %in% colnames(data.nswpcn)) {
    temp.coefs[coef_i] * data.nswpcn[,gsub("TRUE$", "", names(temp.coefs)[coef_i])]
  } else {
    rep(0, nrow(data.nswpcn))
  } })
gg.linpred.nswpcn = -rowSums(gg.linpred.nswpcn) # Negate to bring into concordance with the direction of
temp = summary(fit.gg, newdata = data.nswpcn, ci = FALSE)
gg.prob.nswpcn = sapply(temp, function(x) approx(x[,1], x[,2], xout = val.prob.times, yleft = 1, yright = 1))
colnames(gg.prob.nswpcn) = rownames(data.nswpcn)

```

## 4 Validation

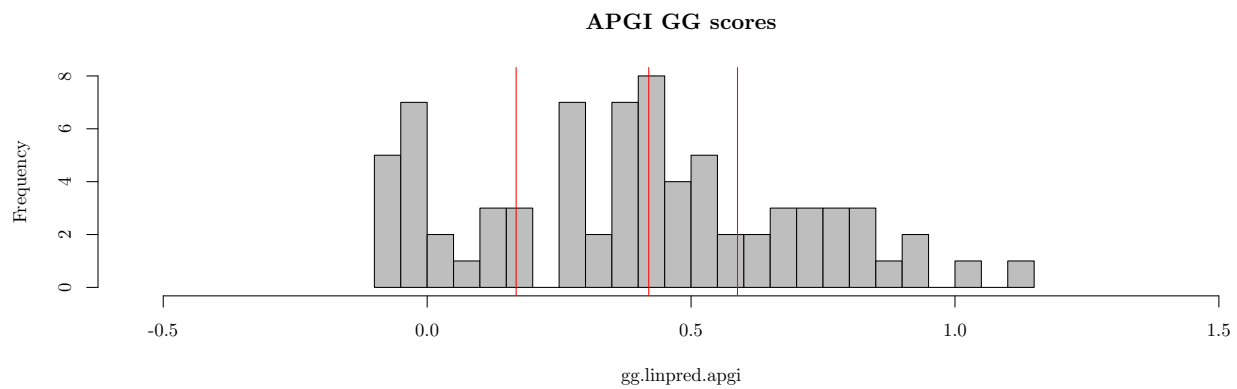
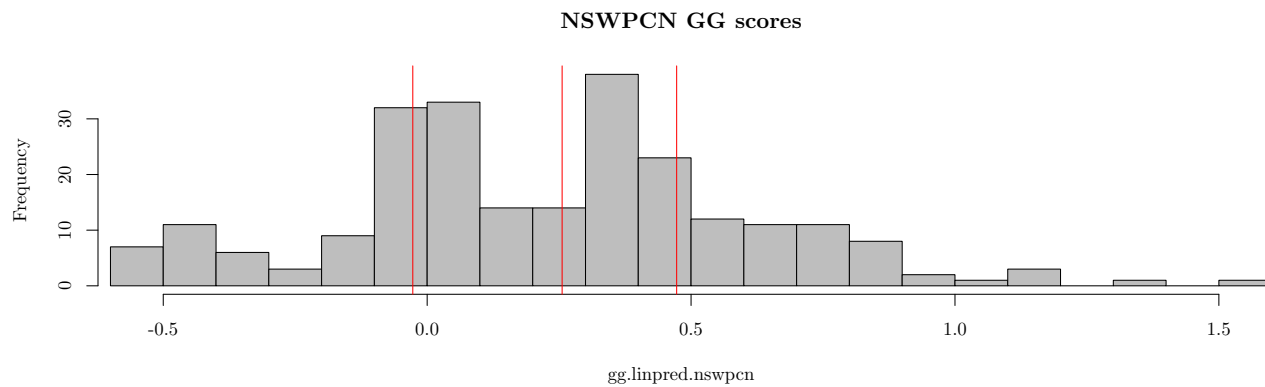
### 4.1 Altman diagnostic 1: score histograms

```

par(mfrow = c(3, 1))
hist(gg.linpred.nswpcn, main = "NSWPCN GG scores", xlim = range(c(gg.linpred.nswpcn, gg.linpred.glasgow)),
     abline(v = quantile(gg.linpred.nswpcn, probs = c(0.25, 0.5, 0.75)), col = "red")
hist(gg.linpred.glasgow, main = "Glasgow GG scores", xlim = range(c(gg.linpred.nswpcn, gg.linpred.glasgow)),
     abline(v = quantile(gg.linpred.glasgow, probs = c(0.25, 0.5, 0.75)), col = "red")
hist(gg.linpred.apgi, main = "APGI GG scores", xlim = range(c(gg.linpred.nswpcn, gg.linpred.glasgow, gg.linpred.apgi)),
     abline(v = quantile(gg.linpred.apgi, probs = c(0.25, 0.5, 0.75)), col = "red")

```





```
par(mfrow = c(1, 1))
```

## 4.2 Altman method 1 (D,F)

```
summary(coxph(Surv(Time, DSD) ~ mskcc_post.linpred.glasgow, data.glasgow))

## Call:
## coxph(formula = Surv(Time, DSD) ~ mskcc_post.linpred.glasgow,
##       data = data.glasgow)
##
##      n= 189, number of events= 161
##
```

```

##               coef exp(coef) se(coef)      z Pr(>|z|)
## mskcc_post.linpred.glasgow 0.01682  1.01696  0.00428 3.93  8.4e-05
##
##               exp(coef) exp(-coef) lower .95 upper .95
## mskcc_post.linpred.glasgow      1.02      0.983      1.01      1.03
##
## Concordance= 0.584 (se = 0.026 )
## Rsquare= 0.081 (max possible= 0.999 )
## Likelihood ratio test= 15.9 on 1 df,  p=6.79e-05
## Wald test              = 15.5 on 1 df,  p=8.43e-05
## Score (logrank) test = 15.7 on 1 df,  p=7.56e-05

summary(coxph(Surv(Time, DSD) ~ mskcc_pre.linpred.glasgow, data.glasgow))

## Call:
## coxph(formula = Surv(Time, DSD) ~ mskcc_pre.linpred.glasgow,
##       data = data.glasgow)
##
## n= 189, number of events= 161
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## mskcc_pre.linpred.glasgow 0.0118  1.0118  0.0105 1.12      0.26
##
##               exp(coef) exp(-coef) lower .95 upper .95
## mskcc_pre.linpred.glasgow      1.01      0.988      0.991      1.03
##
## Concordance= 0.585 (se = 0.026 )
## Rsquare= 0.006 (max possible= 0.999 )
## Likelihood ratio test= 1.15 on 1 df,  p=0.284
## Wald test              = 1.25 on 1 df,  p=0.263
## Score (logrank) test = 1.25 on 1 df,  p=0.264

summary(coxph(Surv(Time, DSD) ~ mskcc_post.linpred.apgi, data.apgi))

## Call:
## coxph(formula = Surv(Time, DSD) ~ mskcc_post.linpred.apgi, data = data.apgi)
##
## n= 75, number of events= 51
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## mskcc_post.linpred.apgi 0.01626  1.01639  0.00452 3.6  0.00032
##
##               exp(coef) exp(-coef) lower .95 upper .95
## mskcc_post.linpred.apgi      1.02      0.984      1.01      1.03
##
## Concordance= 0.701 (se = 0.044 )
## Rsquare= 0.14 (max possible= 0.993 )
## Likelihood ratio test= 11.3 on 1 df,  p=0.000754
## Wald test              = 12.9 on 1 df,  p=0.000319
## Score (logrank) test = 13.3 on 1 df,  p=0.000268

summary(coxph(Surv(Time, DSD) ~ mskcc_pre.linpred.apgi, data.apgi))

## Call:
## coxph(formula = Surv(Time, DSD) ~ mskcc_pre.linpred.apgi, data = data.apgi)

```

```
##
##   n= 75, number of events= 51
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## mskcc_pre.linpred.apgi 0.00329   1.00330  0.00673 0.49    0.62
##
##               exp(coef) exp(-coef) lower .95 upper .95
## mskcc_pre.linpred.apgi      1      0.997    0.99    1.02
##
## Concordance= 0.475  (se = 0.044 )
## Rsquare= 0.003  (max possible= 0.993 )
## Likelihood ratio test= 0.23  on 1 df,  p=0.634
## Wald test            = 0.24  on 1 df,  p=0.625
## Score (logrank) test = 0.24  on 1 df,  p=0.624

summary(coxph(Surv(Time, DSD) ~ gg.linpred.glasgow, data.glasgow))

## Call:
## coxph(formula = Surv(Time, DSD) ~ gg.linpred.glasgow, data = data.glasgow)
##
##   n= 189, number of events= 161
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## gg.linpred.glasgow 0.805     2.236   0.239 3.37  0.00075
##
##               exp(coef) exp(-coef) lower .95 upper .95
## gg.linpred.glasgow     2.24     0.447    1.4    3.57
##
## Concordance= 0.607  (se = 0.026 )
## Rsquare= 0.059  (max possible= 0.999 )
## Likelihood ratio test= 11.4  on 1 df,  p=0.000725
## Wald test            = 11.3  on 1 df,  p=0.000754
## Score (logrank) test = 11.5  on 1 df,  p=0.000705

summary(coxph(Surv(Time, DSD) ~ gg.linpred.apgi, data.apgi))

## Call:
## coxph(formula = Surv(Time, DSD) ~ gg.linpred.apgi, data = data.apgi)
##
##   n= 75, number of events= 51
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## gg.linpred.apgi 1.79     5.99    0.48 3.73  0.00019
##
##               exp(coef) exp(-coef) lower .95 upper .95
## gg.linpred.apgi     5.99    0.167    2.34    15.4
##
## Concordance= 0.645  (se = 0.044 )
## Rsquare= 0.169  (max possible= 0.993 )
## Likelihood ratio test= 13.8  on 1 df,  p=0.000198
## Wald test            = 13.9  on 1 df,  p=0.000194
## Score (logrank) test = 14.3  on 1 df,  p=0.000152

anova(coxph(Surv(Time, DSD) ~ offset(gg.linpred.glasgow) + gg.linpred.glasgow, data.glasgow))
```

```
## Analysis of Deviance Table
## Cox model: response is Surv(Time, DSD)
## Terms added sequentially (first to last)
##
##               loglik Chisq Df Pr(>|Chi|)
## NULL                -678
## gg.linpred.glasgow  -678  0.66  1          0.41

anova(coxph(Surv(Time, DSD) ~ offset(gg.linpred.apgi) + gg.linpred.apgi, data.apgi))

## Analysis of Deviance Table
## Cox model: response is Surv(Time, DSD)
## Terms added sequentially (first to last)
##
##               loglik Chisq Df Pr(>|Chi|)
## NULL                -181
## gg.linpred.apgi    -180  2.71  1          0.099
```

Booyah.

### 4.3 Altman method 2 (F)

```
summary(coxph(Surv(Time, DSD) ~ offset(mskcc_pre.linpred.glasgow) + AgeCent + SexM + SizeCent + A2 + A4,
data.glasgow))

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, : Ran out of
## iterations and did not converge
## Error in fitter(X, Y, strats, offset, init, control, weights = weights, : NA/NaN/Inf in
## foreign function call (arg 6)

summary(coxph(Surv(Time, DSD) ~ offset(mskcc_post.linpred.glasgow) + AgeCent + SexM + SizeCent + A2 + A4,
data.glasgow))

## Call:
## coxph(formula = Surv(Time, DSD) ~ offset(mskcc_post.linpred.glasgow) +
##      AgeCent + SexM + SizeCent + A2 + A4, data = data.glasgow)
##
##      n= 189, number of events= 161
##
##              coef exp(coef)  se(coef)      z Pr(>|z|)
## AgeCent      0.22744   1.25538   0.00862  26.39 < 2e-16
## SexMTRUE     -4.18282   0.01526   0.29544 -14.16 < 2e-16
## SizeCent      0.07140   1.07401   0.01910   3.74 0.00019
## A2TRUE       -2.96537   0.05154   0.41042  -7.23 5e-13
## A4TRUE        5.40464  222.43685   0.28361  19.06 < 2e-16
##
##              exp(coef) exp(-coef) lower .95 upper .95
## AgeCent      1.2554     0.7966  1.23e+00  1.2768
## SexMTRUE      0.0153    65.5506  8.55e-03  0.0272
## SizeCent      1.0740     0.9311  1.03e+00  1.1150
## A2TRUE         0.0515    19.4019  2.31e-02  0.1152
## A4TRUE       222.4369     0.0045  1.28e+02  387.8075
##
## Concordance= 0.588 (se = 0.026 )
## Rsquare= 0.982 (max possible= 1 )
## Likelihood ratio test= 757 on 5 df,  p=0
```

```

## Wald test          = 1654 on 5 df,   p=0
## Score (logrank) test = 1745 on 5 df,   p=0

summary(coxph(Surv(Time, DSD) ~ offset(gg.linpred.glasgow) + AgeCent + SexM + SizeCent + A2 + A4, data.g

## Call:
## coxph(formula = Surv(Time, DSD) ~ offset(gg.linpred.glasgow) +
##       AgeCent + SexM + SizeCent + A2 + A4, data = data.glasgow)
##
## n= 189, number of events= 161
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## AgeCent    -0.03105   0.96943  0.00872 -3.56  0.00037
## SexMTRUE    0.63117   1.87981  0.16671  3.79  0.00015
## SizeCent    0.02245   1.02270  0.00767  2.93  0.00343
## A2TRUE      0.33327   1.39553  0.17564  1.90  0.05776
## A4TRUE     -0.05074   0.95052  0.18482 -0.27  0.78367
##
##              exp(coef) exp(-coef) lower .95 upper .95
## AgeCent         0.969         1.032     0.953     0.986
## SexMTRUE         1.880         0.532     1.356     2.606
## SizeCent         1.023         0.978     1.007     1.038
## A2TRUE           1.396         0.717     0.989     1.969
## A4TRUE           0.951         1.052     0.662     1.365
##
## Concordance= 0.676 (se = 0.026 )
## Rsquare= 0.184 (max possible= 0.999 )
## Likelihood ratio test= 38.4 on 5 df,   p=3.19e-07
## Wald test          = 39 on 5 df,   p=2.4e-07
## Score (logrank) test = 40.5 on 5 df,   p=1.19e-07

summary(coxph(Surv(Time, DSD) ~ offset(mskcc_pre.linpred.apgi) + AgeCent + SexM + SizeCent + A2 + A4, da

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, : Ran out of
iterations and did not converge
## Error in fitter(X, Y, strats, offset, init, control, weights = weights, : NA/NaN/Inf in
foreign function call (arg 6)

summary(coxph(Surv(Time, DSD) ~ offset(mskcc_post.linpred.apgi) + AgeCent + SexM + SizeCent + A2 + A4, c

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, : Ran out of
iterations and did not converge
## Error in fitter(X, Y, strats, offset, init, control, weights = weights, : NA/NaN/Inf in
foreign function call (arg 6)

summary(coxph(Surv(Time, DSD) ~ offset(gg.linpred.apgi) + AgeCent + SexM + SizeCent + A2 + A4, data.apgi

## Warning in coxph(Surv(Time, DSD) ~ offset(gg.linpred.apgi) + AgeCent + SexM + : X matrix
deemed to be singular; variable 2

## Call:
## coxph(formula = Surv(Time, DSD) ~ offset(gg.linpred.apgi) + AgeCent +
##       SexM + SizeCent + A2 + A4, data = data.apgi)
##
## n= 75, number of events= 51
##

```

```
##          coef exp(coef) se(coef)      z Pr(>|z|)
## AgeCent  0.02122   1.02145  0.01775  1.20    0.23
## SexMTRUE      NA         NA  0.00000   NA     NA
## SizeCent  0.01257   1.01265  0.00833  1.51    0.13
## A2TRUE    0.05042   1.05171  0.38919  0.13    0.90
## A4TRUE    0.36722   1.44371  0.32143  1.14    0.25
##
##          exp(coef) exp(-coef) lower .95 upper .95
## AgeCent         1.02      0.979   0.987   1.06
## SexMTRUE         NA         NA     NA     NA
## SizeCent         1.01      0.988   0.996   1.03
## A2TRUE           1.05      0.951   0.490   2.26
## A4TRUE           1.44      0.693   0.769   2.71
##
## Concordance= 0.652  (se = 0.044 )
## Rsquare= 0.064   (max possible= 0.992 )
## Likelihood ratio test= 4.94  on 4 df,   p=0.293
## Wald test              = 4.69  on 4 df,   p=0.32
## Score (logrank) test = 4.74  on 4 df,   p=0.315
```

Still strong evidence of misspecification or poor fit. However, the above calibration slope was not significantly different from 1. Hmm. This doesn't necessarily sink the method, but will need checking as we go along.

## 4.4 Altman method 3 (D)

Look at the CIs above.

## 4.5 Altman method 4 (D,C)

```
group_quantiles = c(0, 0.2, 0.8, 1)
gg.groups.nswpcn = cut(gg.linpred.nswpcn, quantile(gg.linpred.nswpcn, group_quantiles), labels = FALSE)
temp.alpha = 0.1

temp.km = survfit(Surv(data.nswpcn$Time, data.nswpcn$DSD) ~ gg.groups.nswpcn, conf.int = 1-temp.alpha)
temp.km = data.frame(surv = temp.km$surv, group = rep(gsub(".*=", "", names(temp.km$strata)), temp.km$strata))
temp.pred = summary(fit.gg, newdata = data.nswpcn, ci = FALSE)
temp.pred.times = temp.pred[[1]][,1]
temp.pred.ests = sapply(temp.pred, function(x) x[,2])
temp.pred.ests = tapply(1:ncol(temp.pred.ests), gg.groups.nswpcn, function(is) apply(temp.pred.ests[,is], 1, FUN = function(x) x[is,]))
temp.pred.lower = sapply(temp.pred.ests, function(x) x[1,])
temp.pred.meds = sapply(temp.pred.ests, function(x) x[2,])
temp.pred.upper = sapply(temp.pred.ests, function(x) x[3,])
temp.pred = data.frame(surv = as.vector(temp.pred.meds), group = rep(colnames(temp.pred.meds), each = nrow(temp.pred.meds)))
temp.data = rbind(temp.km, temp.pred)
# ggplot(temp.data, aes(x = time, y = surv, colour = group, fill = group, ymax = upper, ymin = lower, lty = 1)) +
#   geom_step() +
#   xlim(0, 5*365) +
#   labs(title = "Goodness of fit: model GG1 on NSWPCN training data", x = "Time from diagnosis (days)", y = "Survival probability")

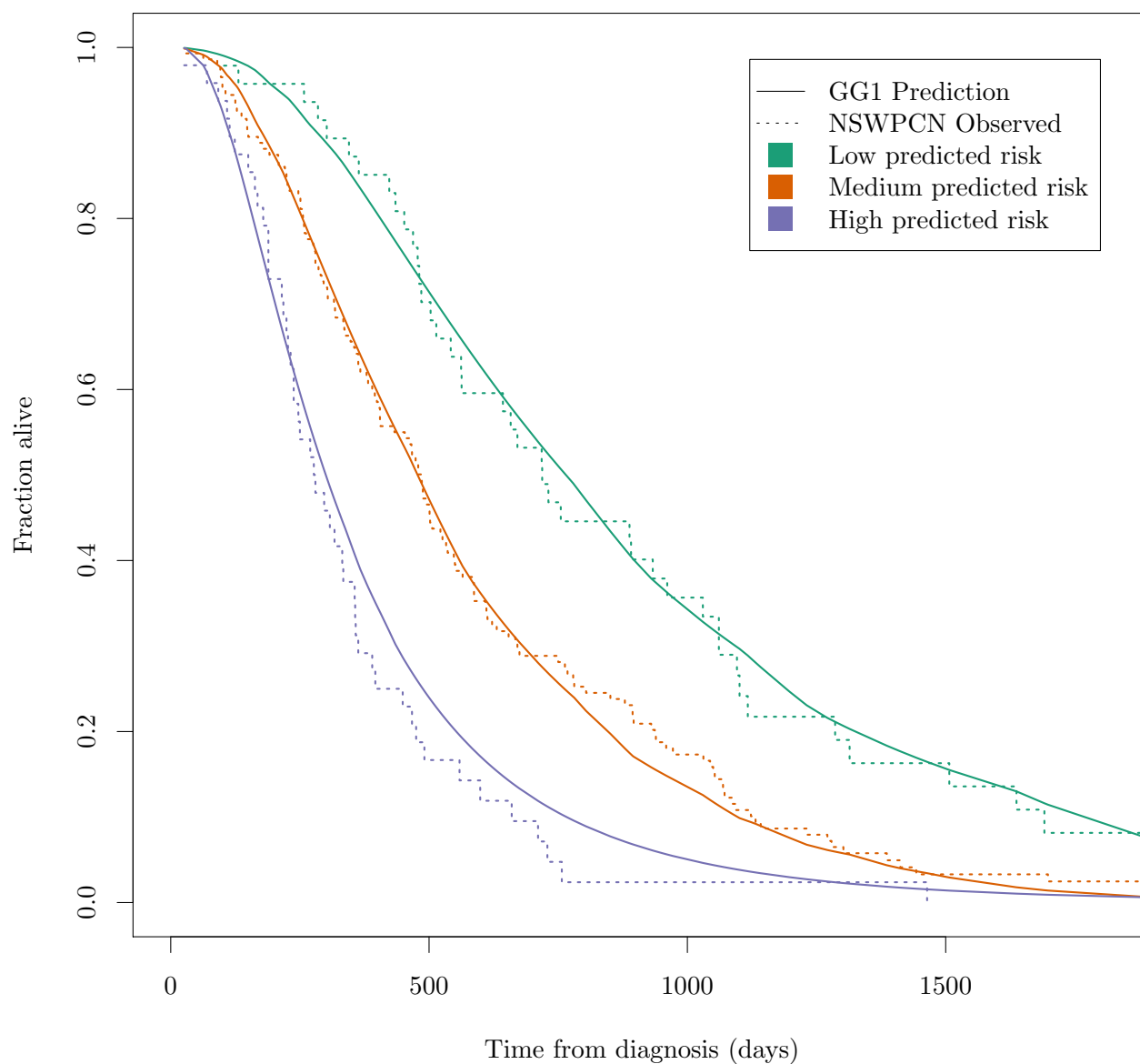
plot(0 ~ 0, type = "n", xlim = c(0, 5*365), ylim = c(0, 1), main = "Goodness of fit: model GG1 on NSWPCN training data")
temp.pal = brewer.pal(length(unique(gg.groups.nswpcn)), "Dark2")
```

```

names(temp.pal) = sort(unique(gg.groups.nswpcn))
for (temp.i in factor(sort(unique(gg.groups.nswpcn))))
{
  lines(surv ~ time, temp.data[as.character(temp.data$group) == as.character(temp.i) & temp.data$group == as.character(temp.i)], col = temp.pal[temp.i])
  lines(surv ~ time, temp.data[as.character(temp.data$group) == as.character(temp.i) & temp.data$group == as.character(temp.i)], col = temp.pal[temp.i])
}
legend("topright", inset = 0.05, legend = c("GG1 Prediction", "NSWPCN Observed", "Low predicted risk", "Medium predicted risk", "High predicted risk"))

```

### Goodness of fit: model GG1 on NSWPCN training data



```

summary(coxph(Surv(data.nswpcn$Time, data.nswpcn$DSD) ~ factor(gg.groups.nswpcn)))
## Call:
## coxph(formula = Surv(data.nswpcn$Time, data.nswpcn$DSD) ~ factor(gg.groups.nswpcn))
##
## n= 239, number of events= 230

```

```
##      (1 observation deleted due to missingness)
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## factor(gg.groups.nswpcn)2 0.532    1.703    0.176 3.03  0.0025
## factor(gg.groups.nswpcn)3 1.328    3.775    0.219 6.06  1.3e-09
##
##              exp(coef) exp(-coef) lower .95 upper .95
## factor(gg.groups.nswpcn)2    1.70    0.587    1.21    2.4
## factor(gg.groups.nswpcn)3    3.78    0.265    2.46    5.8
##
## Concordance= 0.618  (se = 0.019 )
## Rsquare= 0.138  (max possible= 1 )
## Likelihood ratio test= 35.5  on 2 df,  p=1.96e-08
## Wald test              = 37.9  on 2 df,  p=6.01e-09
## Score (logrank) test = 40.7  on 2 df,  p=1.46e-09
```

```
mskcc_pre.groups.glasgow = cut(mskcc_pre.linpred.glasgow, quantile(mskcc_pre.linpred.glasgow, group_quantiles), labels = FALSE)
mskcc_post.groups.glasgow = cut(mskcc_post.linpred.glasgow, quantile(mskcc_post.linpred.glasgow, group_quantiles), labels = FALSE)
gg.groups.glasgow = cut(gg.linpred.glasgow, quantile(gg.linpred.glasgow, group_quantiles), labels = FALSE)

temp.km = survfit(Surv(data.glasgow$Time, data.glasgow$DSD) ~ gg.groups.glasgow, conf.int = 1-temp.alpha)
temp.km = data.frame(surv = temp.km$surv, group = rep(gsub(".*=", "", names(temp.km$strata)), temp.km$strata))
temp.pred = summary(fit.gg, newdata = data.glasgow, ci = FALSE)
temp.pred.times = temp.pred[[1]][,1]
temp.pred.ests = sapply(temp.pred, function(x) x[,2])
temp.pred.ests = tapply(1:ncol(temp.pred.ests), gg.groups.glasgow, function(is) apply(temp.pred.ests[,is], 1, FUN = function(x) x[is,]))
temp.pred.lower = sapply(temp.pred.ests, function(x) x[1,])
temp.pred.meds = sapply(temp.pred.ests, function(x) x[2,])
temp.pred.upper = sapply(temp.pred.ests, function(x) x[3,])
temp.pred = data.frame(surv = as.vector(temp.pred.meds), group = rep(colnames(temp.pred.meds), each = nrow(temp.pred.meds)))
temp.data = rbind(temp.km, temp.pred)

temp.predpre.12mo = simplify2array(tapply(mskcc_pre.12mo.glasgow, mskcc_pre.groups.glasgow, quantile, probs = 0.95))
temp.predpre.24mo = simplify2array(tapply(mskcc_pre.24mo.glasgow, mskcc_pre.groups.glasgow, quantile, probs = 0.95))
temp.predpre.36mo = simplify2array(tapply(mskcc_pre.36mo.glasgow, mskcc_pre.groups.glasgow, quantile, probs = 0.95))
temp.predpost.12mo = simplify2array(tapply(mskcc_post.12mo.glasgow, mskcc_post.groups.glasgow, quantile, probs = 0.95))
temp.predpost.24mo = simplify2array(tapply(mskcc_post.24mo.glasgow, mskcc_post.groups.glasgow, quantile, probs = 0.95))
temp.predpost.36mo = simplify2array(tapply(mskcc_post.36mo.glasgow, mskcc_post.groups.glasgow, quantile, probs = 0.95))
temp.data2 = data.frame(
  surv = c(temp.predpre.12mo[2,], temp.predpre.24mo[2,], temp.predpre.36mo[2,], temp.predpost.12mo[2,], temp.predpost.24mo[2,], temp.predpost.36mo[2,]),
  group = factor(rep(sort(unique(mskcc_pre.groups.glasgow)), 6)),
  time = rep(c(12, 24, 36)/12*365.25, each = 3),
  upper = c(temp.predpre.12mo[3,], temp.predpre.24mo[3,], temp.predpre.36mo[3,], temp.predpost.12mo[3,], temp.predpost.24mo[3,], temp.predpost.36mo[3,]),
  lower = c(temp.predpre.12mo[1,], temp.predpre.24mo[1,], temp.predpre.36mo[1,], temp.predpost.12mo[1,], temp.predpost.24mo[1,], temp.predpost.36mo[1,]),
  est = rep(c("MSKCC Preoperative", "MSKCC Postoperative"), each = 9))

# ggplot(temp.data, aes(x = time, y = surv, colour = group, fill = group, ymax = upper, ymin = lower, lty = 1)) +
#   geom_step() +
#   xlim(0, 5*365) +
#   geom_line(data = temp.data2) +
#   labs(title = "Goodness of fit: model GG1 on Glasgow validation data", x = "Time from diagnosis (days)")

plot(0 ~ 0, type = "n", xlim = c(0, 5*365), ylim = c(0, 1), main = "Goodness of fit: model GG1 on Glasgow validation data")
temp.pal = brewer.pal(length(unique(gg.groups.glasgow)), "Dark2")
names(temp.pal) = sort(unique(gg.groups.glasgow))
```

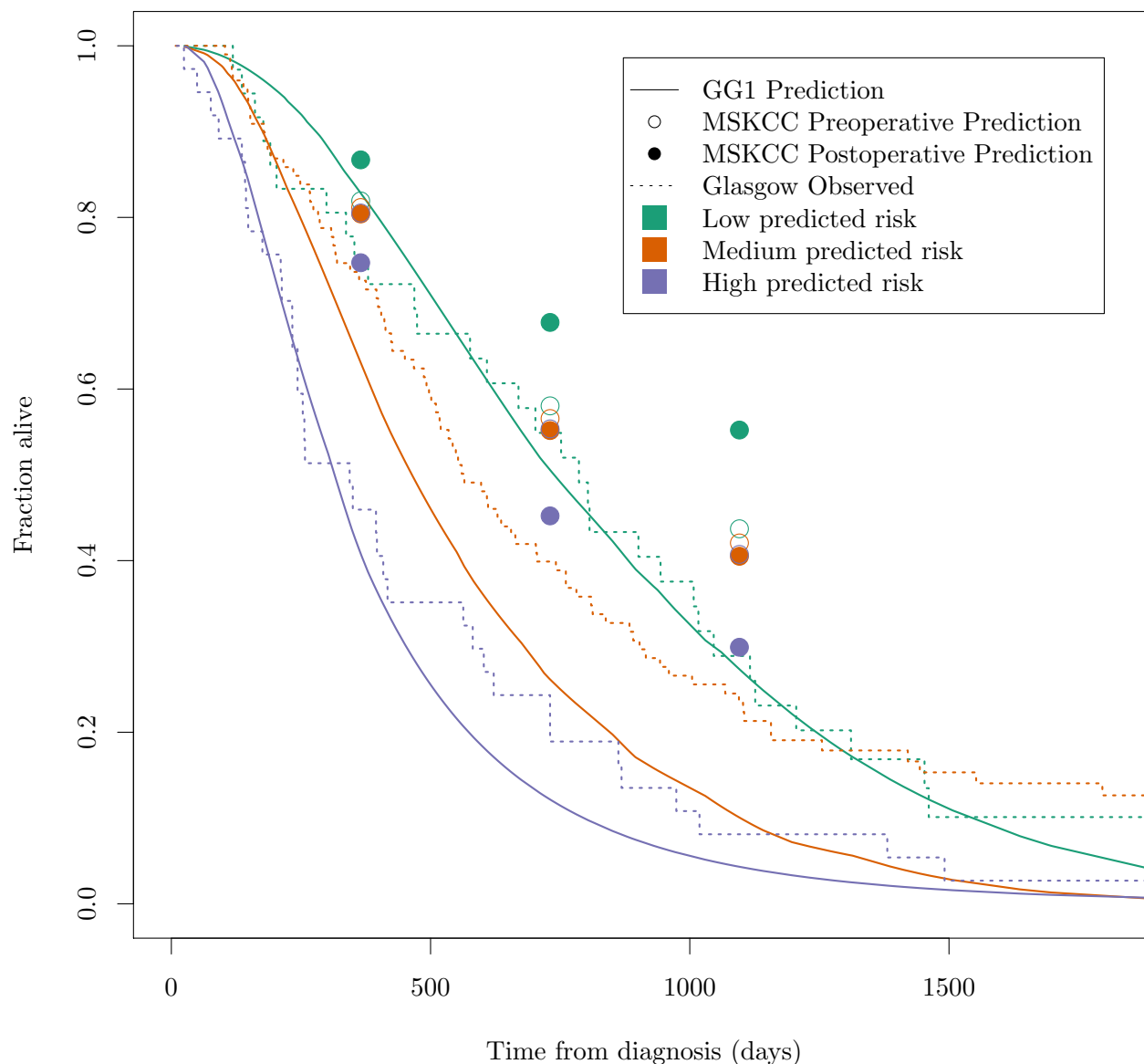


```

for (temp.i in factor(sort(unique(gg.groups.glasgow))))
{
  lines(surv ~ time, temp.data[as.character(temp.data$group) == as.character(temp.i) & temp.data$group == as.character(temp.i)], col = "black", lty = 1)
  lines(surv ~ time, temp.data[as.character(temp.data$group) == as.character(temp.i) & temp.data$group == as.character(temp.i)], col = "black", lty = 1)
  points(surv ~ time, temp.data2[as.character(temp.data2$group) == as.character(temp.i) & temp.data2$group == as.character(temp.i)], col = "black", lty = 1)
  points(surv ~ time, temp.data2[as.character(temp.data2$group) == as.character(temp.i) & temp.data2$group == as.character(temp.i)], col = "black", lty = 1)
}
legend("topright", inset = 0.05, legend = c("GG1 Prediction", "MSKCC Preoperative Prediction", "MSKCC Postoperative Prediction", "Glasgow Observed", "Low predicted risk", "Medium predicted risk", "High predicted risk"))

```

## Goodness of fit: model GG1 on Glasgow validation data



```

summary(coxph(Surv(data.glasgow$Time, data.glasgow$DSD) ~ factor(gg.groups.glasgow)))
## Call:
## coxph(formula = Surv(data.glasgow$Time, data.glasgow$DSD) ~ factor(gg.groups.glasgow))
##

```

```
## n= 188, number of events= 160
## (1 observation deleted due to missingness)
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## factor(gg.groups.glasgow)2 0.0794    1.0826    0.2074 0.38    0.7019
## factor(gg.groups.glasgow)3 0.6662    1.9468    0.2438 2.73    0.0063
##
##               exp(coef) exp(-coef) lower .95 upper .95
## factor(gg.groups.glasgow)2      1.08      0.924      0.721      1.63
## factor(gg.groups.glasgow)3      1.95      0.514      1.207      3.14
##
## Concordance= 0.577 (se = 0.023 )
## Rsquare= 0.049 (max possible= 0.999 )
## Likelihood ratio test= 9.37 on 2 df, p=0.00923
## Wald test = 10.4 on 2 df, p=0.00543
## Score (logrank) test = 10.8 on 2 df, p=0.00463

summary(coxph(Surv(data.glasgow$Time, data.glasgow$DSD) ~ factor(mskcc_pre.groups.glasgow)))

## Call:
## coxph(formula = Surv(data.glasgow$Time, data.glasgow$DSD) ~ factor(mskcc_pre.groups.glasgow))
##
## n= 188, number of events= 160
## (1 observation deleted due to missingness)
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## factor(mskcc_pre.groups.glasgow)2 0.764      2.147    0.217 3.52    0.00043
## factor(mskcc_pre.groups.glasgow)3 0.762      2.143    0.260 2.93    0.00338
##
##               exp(coef) exp(-coef) lower .95 upper .95
## factor(mskcc_pre.groups.glasgow)2      2.15      0.466      1.40      3.28
## factor(mskcc_pre.groups.glasgow)3      2.14      0.467      1.29      3.57
##
## Concordance= 0.563 (se = 0.023 )
## Rsquare= 0.077 (max possible= 0.999 )
## Likelihood ratio test= 15.1 on 2 df, p=0.000535
## Wald test = 13.1 on 2 df, p=0.00144
## Score (logrank) test = 13.6 on 2 df, p=0.00109

summary(coxph(Surv(data.glasgow$Time, data.glasgow$DSD) ~ factor(mskcc_post.groups.glasgow)))

## Call:
## coxph(formula = Surv(data.glasgow$Time, data.glasgow$DSD) ~ factor(mskcc_post.groups.glasgow))
##
## n= 188, number of events= 160
## (1 observation deleted due to missingness)
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## factor(mskcc_post.groups.glasgow)2 0.631      1.879    0.218 2.9    0.00378
## factor(mskcc_post.groups.glasgow)3 0.990      2.691    0.261 3.8    0.00015
##
##               exp(coef) exp(-coef) lower .95
## factor(mskcc_post.groups.glasgow)2      1.88      0.532      1.23
## factor(mskcc_post.groups.glasgow)3      2.69      0.372      1.61
##               upper .95
```

```
## factor(mskcc_post.groups.glasgow)2      2.88
## factor(mskcc_post.groups.glasgow)3      4.49
##
## Concordance= 0.579 (se = 0.023 )
## Rsquare= 0.081 (max possible= 0.999 )
## Likelihood ratio test= 15.8 on 2 df, p=0.000372
## Wald test = 14.7 on 2 df, p=0.00066
## Score (logrank) test = 15.3 on 2 df, p=0.000484
```

```
mskcc_pre.groups.apgi = cut(mskcc_pre.linpred.apgi, quantile(mskcc_pre.linpred.apgi, group_quantiles), 1)
mskcc_post.groups.apgi = cut(mskcc_post.linpred.apgi, quantile(mskcc_post.linpred.apgi, group_quantiles), 1)
gg.groups.apgi = cut(gg.linpred.apgi, quantile(gg.linpred.apgi, group_quantiles), labels = FALSE)

temp.km = survfit(Surv(data.apgi$Time, data.apgi$DSD) ~ gg.groups.apgi, conf.int = 1-temp.alpha)
temp.km = data.frame(surv = temp.km$surv, group = rep(gsub(".*=", "", names(temp.km$strata)), temp.km$strata))
temp.pred = summary(fit.gg, newdata = data.apgi, ci = FALSE)
temp.pred.times = temp.pred[[1]][,1]
temp.pred.ests = sapply(temp.pred, function(x) x[,2])
temp.pred.ests = tapply(1:ncol(temp.pred.ests), gg.groups.apgi, function(is) apply(temp.pred.ests[,is], 1, function(x) x[is,]))
temp.pred.lower = sapply(temp.pred.ests, function(x) x[1,])
temp.pred.meds = sapply(temp.pred.ests, function(x) x[2,])
temp.pred.upper = sapply(temp.pred.ests, function(x) x[3,])
temp.pred = data.frame(surv = as.vector(temp.pred.meds), group = rep(colnames(temp.pred.meds), each = nrow(temp.pred.meds)))
temp.data = rbind(temp.km, temp.pred)

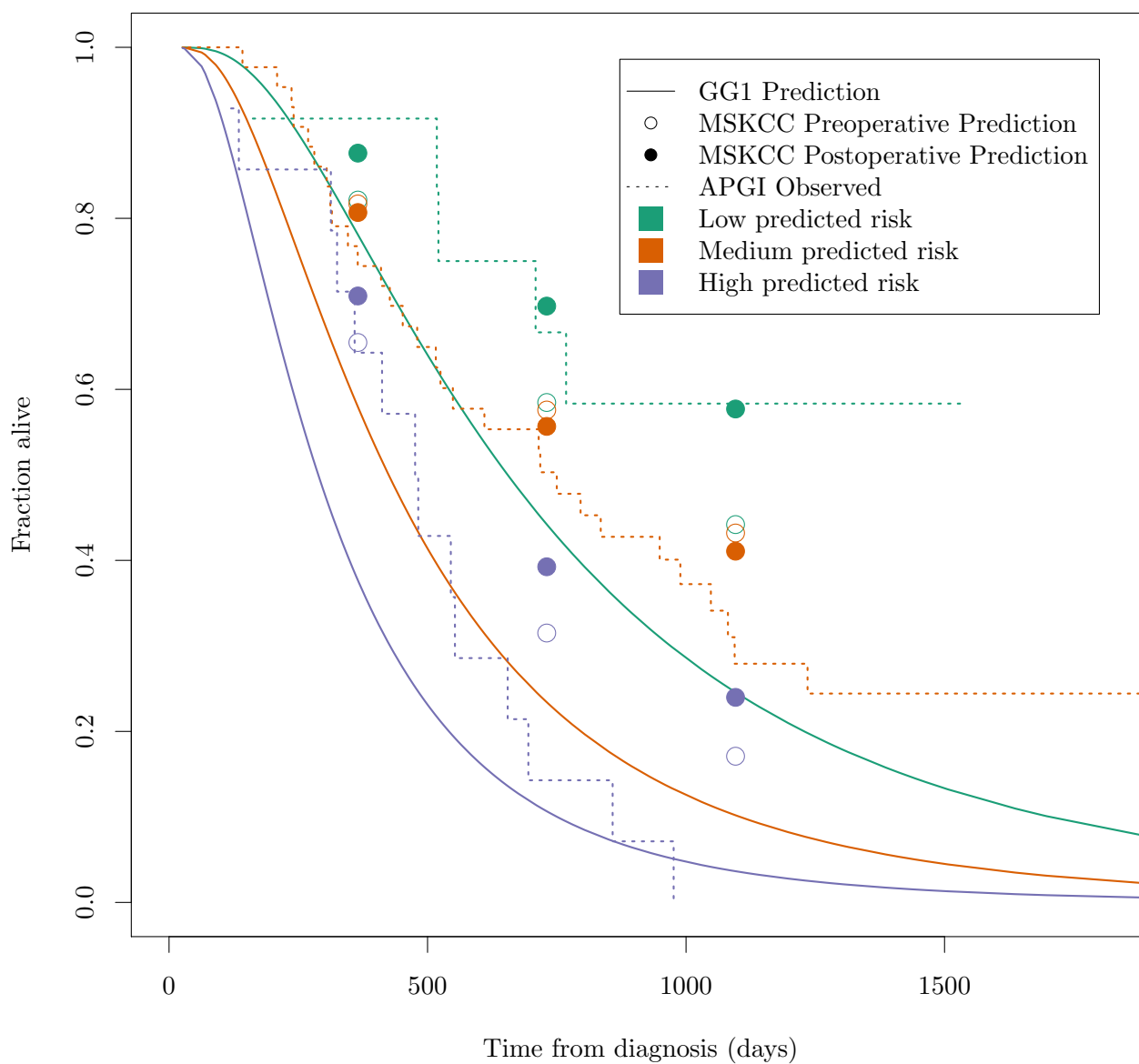
temp.predpre.12mo = simplify2array(tapply(mskcc_pre.12mo.apgi, mskcc_pre.groups.apgi, quantile, probs = 0.25))
temp.predpre.24mo = simplify2array(tapply(mskcc_pre.24mo.apgi, mskcc_pre.groups.apgi, quantile, probs = 0.25))
temp.predpre.36mo = simplify2array(tapply(mskcc_pre.36mo.apgi, mskcc_pre.groups.apgi, quantile, probs = 0.25))
temp.predpost.12mo = simplify2array(tapply(mskcc_post.12mo.apgi, mskcc_post.groups.apgi, quantile, probs = 0.25))
temp.predpost.24mo = simplify2array(tapply(mskcc_post.24mo.apgi, mskcc_post.groups.apgi, quantile, probs = 0.25))
temp.predpost.36mo = simplify2array(tapply(mskcc_post.36mo.apgi, mskcc_post.groups.apgi, quantile, probs = 0.25))
temp.data2 = data.frame(
  surv = c(temp.predpre.12mo[2,], temp.predpre.24mo[2,], temp.predpre.36mo[2,], temp.predpost.12mo[2,], temp.predpost.24mo[2,], temp.predpost.36mo[2,]),
  group = factor(rep(sort(unique(mskcc_pre.groups.apgi)), 6)),
  time = rep(c(12, 24, 36)/12*365.25, each = 3),
  upper = c(temp.predpre.12mo[3,], temp.predpre.24mo[3,], temp.predpre.36mo[3,], temp.predpost.12mo[3,], temp.predpost.24mo[3,], temp.predpost.36mo[3,]),
  lower = c(temp.predpre.12mo[1,], temp.predpre.24mo[1,], temp.predpre.36mo[1,], temp.predpost.12mo[1,], temp.predpost.24mo[1,], temp.predpost.36mo[1,]),
  est = rep(c("MSKCC Preoperative", "MSKCC Postoperative"), each = 9))

# ggplot(temp.data, aes(x = time, y = surv, colour = group, fill = group, ymax = upper, ymin = lower, lty = 1)) +
#   geom_step() +
#   xlim(0, 5*365) +
#   geom_line(data = temp.data2) +
#   labs(title = "Goodness of fit: model GG1 on APGI validation data", x = "Time from diagnosis (days)", y = "Survival")

plot(0 ~ 0, type = "n", xlim = c(0, 5*365), ylim = c(0, 1), main = "Goodness of fit: model GG1 on APGI validation data")
temp.pal = brewer.pal(length(unique(gg.groups.apgi)), "Dark2")
names(temp.pal) = sort(unique(gg.groups.apgi))
for (temp.i in factor(sort(unique(gg.groups.apgi))))
{
  lines(surv ~ time, temp.data[as.character(temp.data$group) == as.character(temp.i) & temp.data$group == as.character(temp.i)], col = temp.pal[temp.i])
  lines(surv ~ time, temp.data[as.character(temp.data$group) == as.character(temp.i) & temp.data$group != as.character(temp.i)], col = "black", lty = 2)
  points(surv ~ time, temp.data2[as.character(temp.data2$group) == as.character(temp.i) & temp.data2$group == as.character(temp.i)], col = temp.pal[temp.i])
  points(surv ~ time, temp.data2[as.character(temp.data2$group) == as.character(temp.i) & temp.data2$group != as.character(temp.i)], col = "black", lty = 2)
}
```

```
legend("topright", inset = 0.05, legend = c("GG1 Prediction", "MSKCC Preoperative Prediction", "MSKCC Postoperative Prediction", "APGI Observed", "Low predicted risk", "Medium predicted risk", "High predicted risk"))
```

### Goodness of fit: model GG1 on APGI validation data



```
summary(coxph(Surv(data.apgi$Time, data.apgi$DSD) ~ factor(gg.groups.apgi)))

## Call:
## coxph(formula = Surv(data.apgi$Time, data.apgi$DSD) ~ factor(gg.groups.apgi))
##
## n= 72, number of events= 50
## (3 observations deleted due to missingness)
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## factor(gg.groups.apgi)2  0.784    2.190   0.484  1.62  0.1051
## factor(gg.groups.apgi)3  1.689    5.413   0.533  3.17  0.0015
##
```

```

##               exp(coef) exp(-coef) lower .95 upper .95
## factor(gg.groups.apgi)2      2.19      0.457      0.849      5.65
## factor(gg.groups.apgi)3      5.41      0.185      1.905     15.38
##
## Concordance= 0.609 (se = 0.039 )
## Rsquare= 0.153 (max possible= 0.993 )
## Likelihood ratio test= 11.9 on 2 df, p=0.00254
## Wald test = 12 on 2 df, p=0.00249
## Score (logrank) test = 13.4 on 2 df, p=0.00124

summary(coxph(Surv(data.apgi$Time, data.apgi$DSD) ~ factor(mskcc_pre.groups.apgi)))

## Call:
## coxph(formula = Surv(data.apgi$Time, data.apgi$DSD) ~ factor(mskcc_pre.groups.apgi))
##
## n= 74, number of events= 50
## (1 observation deleted due to missingness)
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## factor(mskcc_pre.groups.apgi)2 -0.412      0.662      0.367 -1.12      0.26
## factor(mskcc_pre.groups.apgi)3 -0.058      0.944      0.449 -0.13      0.90
##
##               exp(coef) exp(-coef) lower .95 upper .95
## factor(mskcc_pre.groups.apgi)2      0.662      1.51      0.322      1.36
## factor(mskcc_pre.groups.apgi)3      0.944      1.06      0.392      2.27
##
## Concordance= 0.559 (se = 0.037 )
## Rsquare= 0.023 (max possible= 0.993 )
## Likelihood ratio test= 1.7 on 2 df, p=0.428
## Wald test = 1.75 on 2 df, p=0.417
## Score (logrank) test = 1.77 on 2 df, p=0.412

summary(coxph(Surv(data.apgi$Time, data.apgi$DSD) ~ factor(mskcc_post.groups.apgi)))

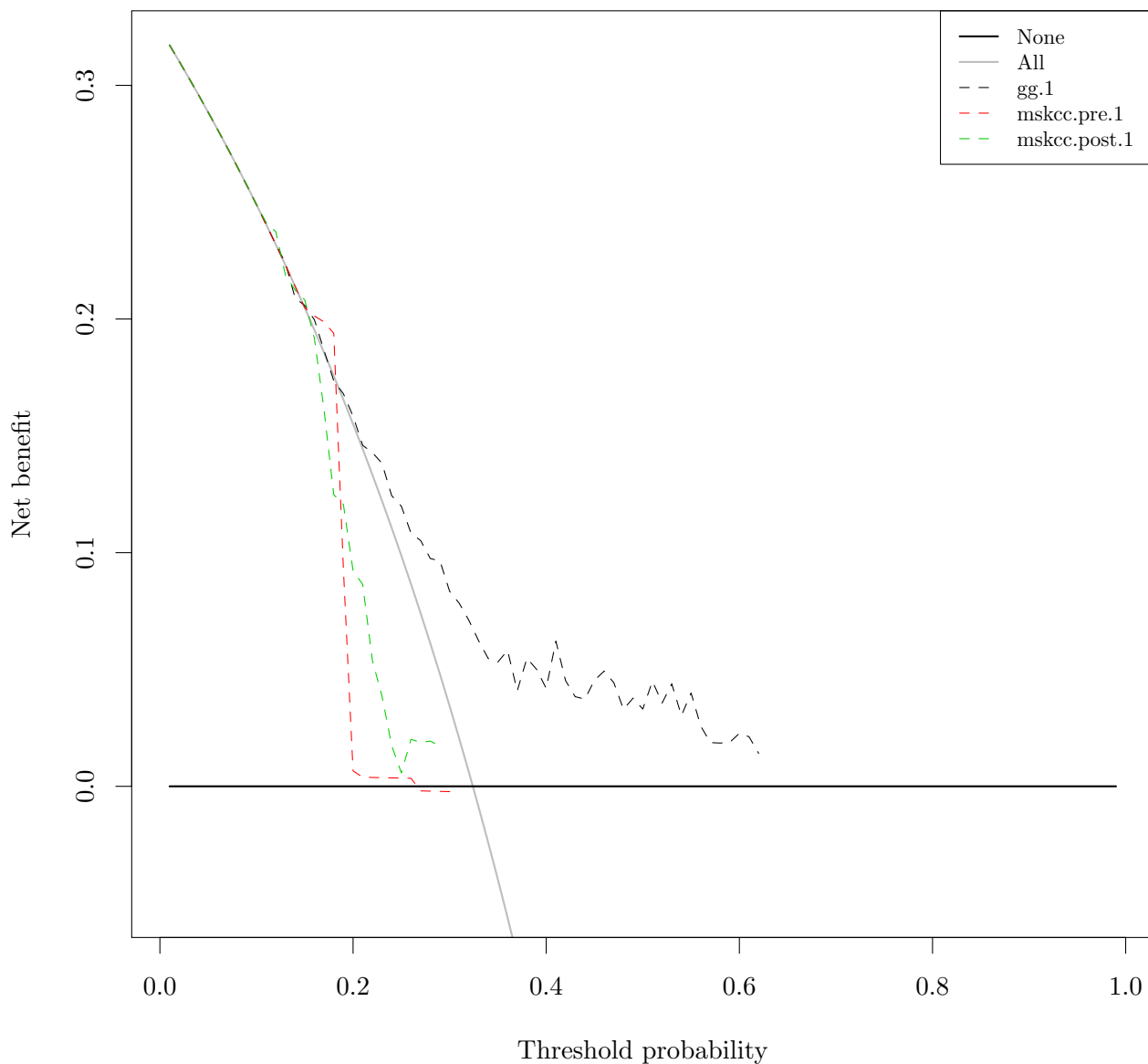
## Call:
## coxph(formula = Surv(data.apgi$Time, data.apgi$DSD) ~ factor(mskcc_post.groups.apgi))
##
## n= 74, number of events= 51
## (1 observation deleted due to missingness)
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## factor(mskcc_post.groups.apgi)2 1.526      4.598      0.531 2.87      0.0041
## factor(mskcc_post.groups.apgi)3 1.812      6.125      0.576 3.15      0.0016
##
##               exp(coef) exp(-coef) lower .95 upper .95
## factor(mskcc_post.groups.apgi)2      4.60      0.217      1.62     13.0
## factor(mskcc_post.groups.apgi)3      6.12      0.163      1.98     18.9
##
## Concordance= 0.624 (se = 0.04 )
## Rsquare= 0.184 (max possible= 0.993 )
## Likelihood ratio test= 15.1 on 2 df, p=0.000539
## Wald test = 10.1 on 2 df, p=0.00628
## Score (logrank) test = 12.3 on 2 df, p=0.00208

```

Decision curve analysis.

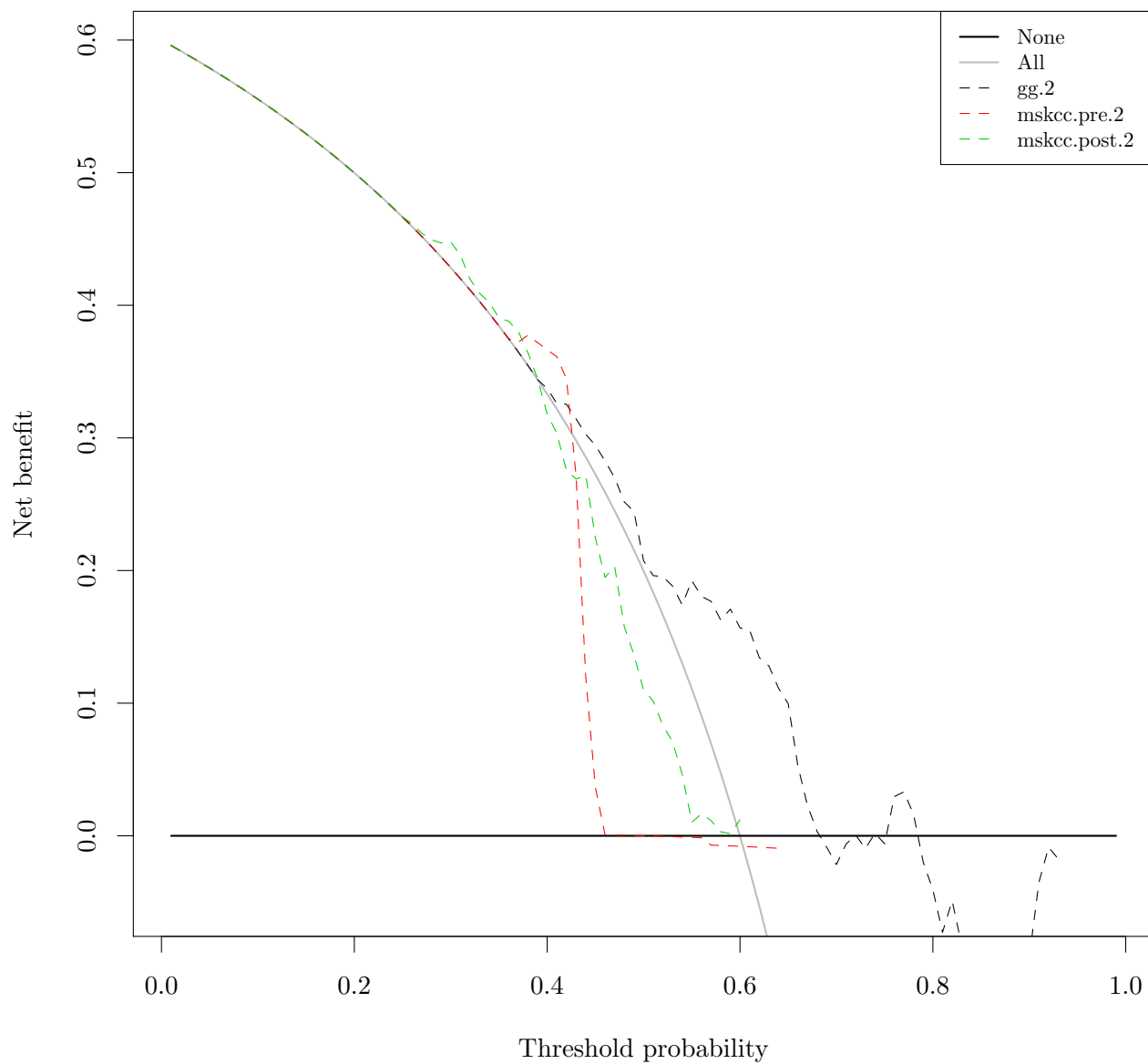
```
source("stdca.R")
temp.data = data.frame(Time = data.glasgow$Time, DSD = data.glasgow$DSD*1,
  gg.1 = 1-gg.prob.glasgow[val.prob.times == 365,], gg.2 = 1-gg.prob.glasgow[val.prob.times == 365*2,],
  mskcc.pre.1 = 1-mskcc_pre.12mo.glasgow, mskcc.pre.2 = 1-mskcc_pre.24mo.glasgow, mskcc.pre.3 = 1-mskcc_pre.36mo.glasgow,
  mskcc.post.1 = 1-mskcc_post.12mo.glasgow, mskcc.post.2 = 1-mskcc_post.24mo.glasgow, mskcc.post.3 = 1-mskcc_post.36mo.glasgow)
invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.1", "mskcc.pre.1", "mskcc.post.1")))

## [1] "gg.1: No observations with risk greater than 63% that have followup through the timepoint selected"
## [2] "mskcc.pre.1: No observations with risk greater than 32%, and therefore net benefit not calculable"
## [3] "mskcc.post.1: No observations with risk greater than 30% that have followup through the timepoint selected"
```

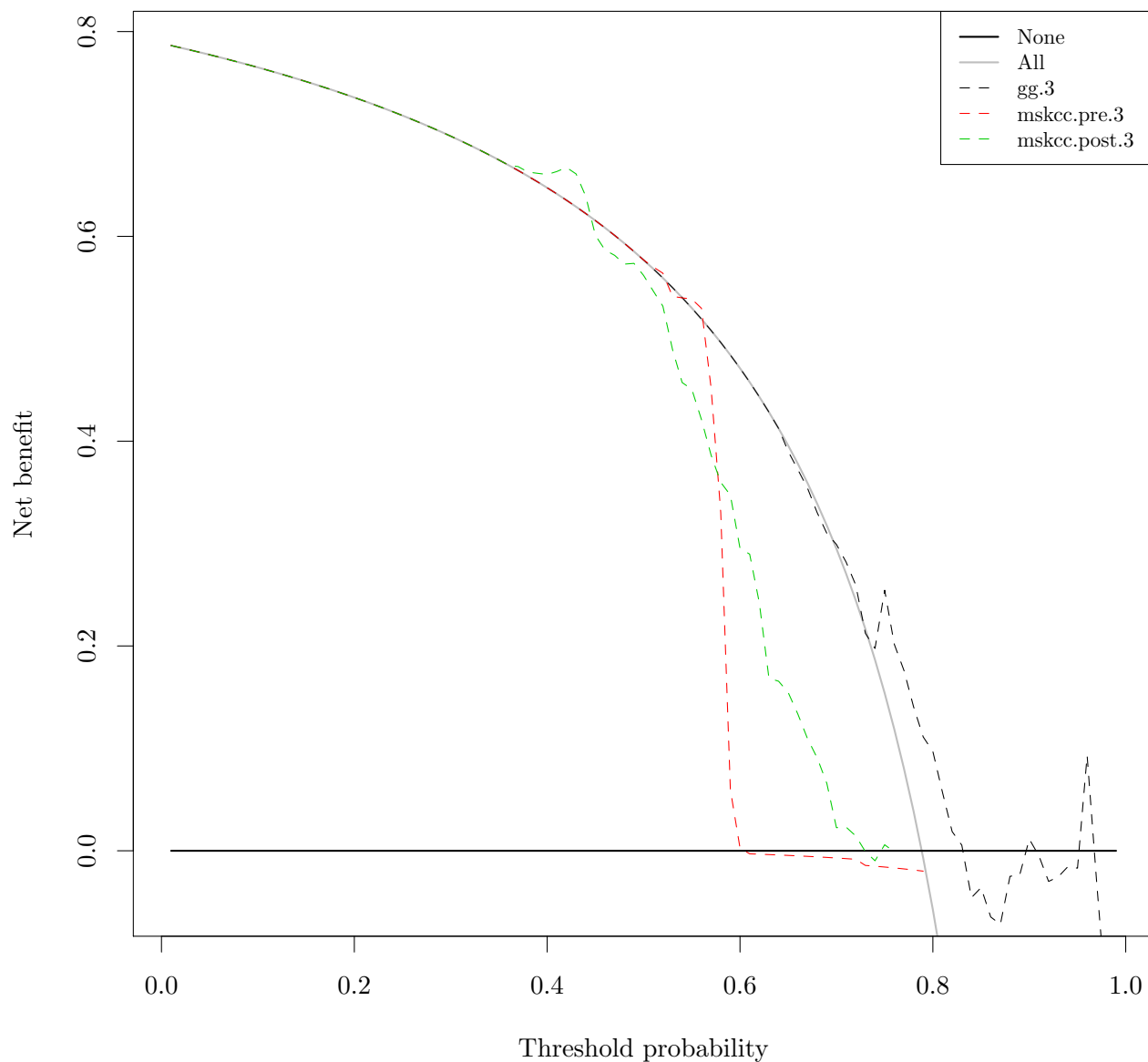


```
invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.2", "mskcc.pre.2", "mskcc.post.2")))

## [1] "gg.2: No observations with risk greater than 94% that have followup through the timepoint selected"
## [2] "mskcc.pre.2: No observations with risk greater than 65%, and therefore net benefit not calculable"
## [3] "mskcc.post.2: No observations with risk greater than 61% that have followup through the timepoint selected"
```



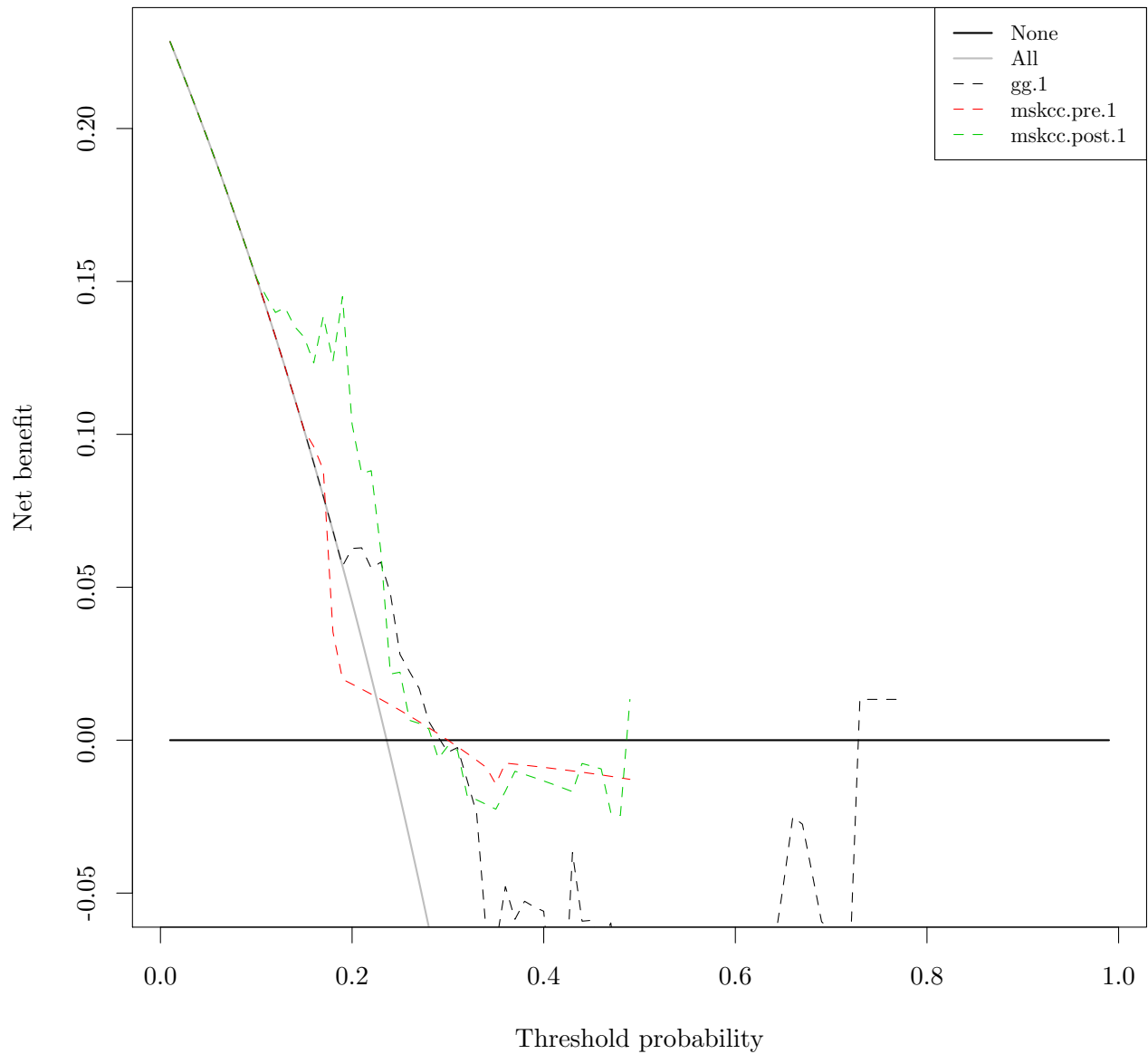
```
invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.3", "mskcc.pre.3", "mskcc.post.3")))
## [1] "mskcc.pre.3: No observations with risk greater than 80%, and therefore net benefit not calculabl"
## [2] "mskcc.post.3: No observations with risk greater than 77% that have followup through the timepoint"
```



```
temp.data = data.frame(Time = data.apgi$Time, DSD = data.apgi$DSD*1,
  gg.1 = 1-gg.prob.apgi[val.prob.times == 365,], gg.2 = 1-gg.prob.apgi[val.prob.times == 365*2,], gg.3 = 1-gg.prob.apgi[val.prob.times == 365*3,],
  mskcc.pre.1 = 1-mskcc_pre.12mo.apgi, mskcc.pre.2 = 1-mskcc_pre.24mo.apgi, mskcc.pre.3 = 1-mskcc_pre.36mo.apgi,
  mskcc.post.1 = 1-mskcc_post.12mo.apgi, mskcc.post.2 = 1-mskcc_post.24mo.apgi, mskcc.post.3 = 1-mskcc_post.36mo.apgi,
  invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.1", "mskcc.pre.1", "mskcc.post.1", "gg.2", "mskcc.pre.2", "mskcc.post.2", "gg.3", "mskcc.pre.3", "mskcc.post.3")))

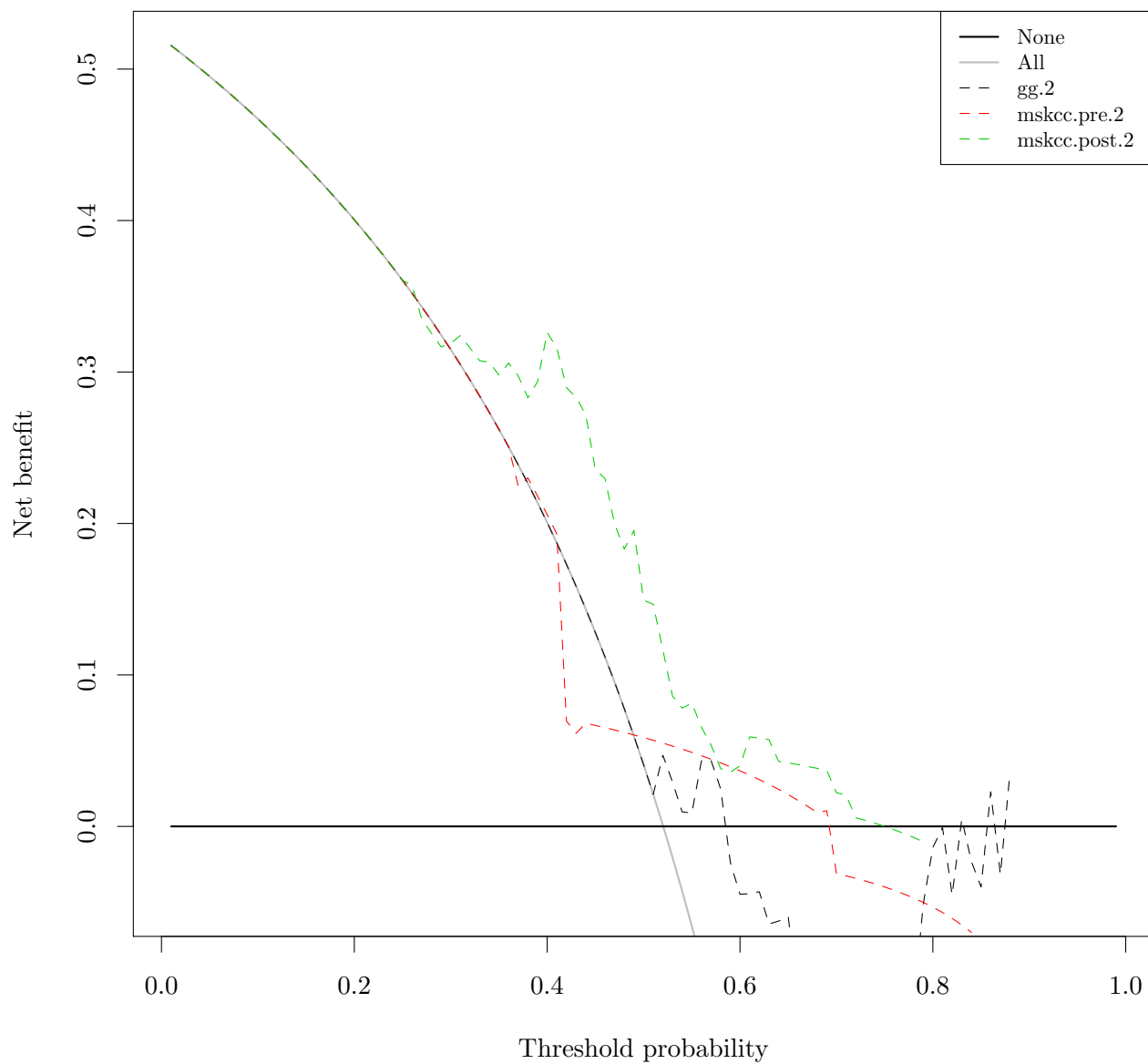
## [1] "gg.1: No observations with risk greater than 78%, and therefore net benefit not calculable in the threshold range"
## [2] "mskcc.pre.1: No observations with risk greater than 50%, and therefore net benefit not calculable in the threshold range"
## [3] "mskcc.post.1: No observations with risk greater than 50%, and therefore net benefit not calculable in the threshold range"
```





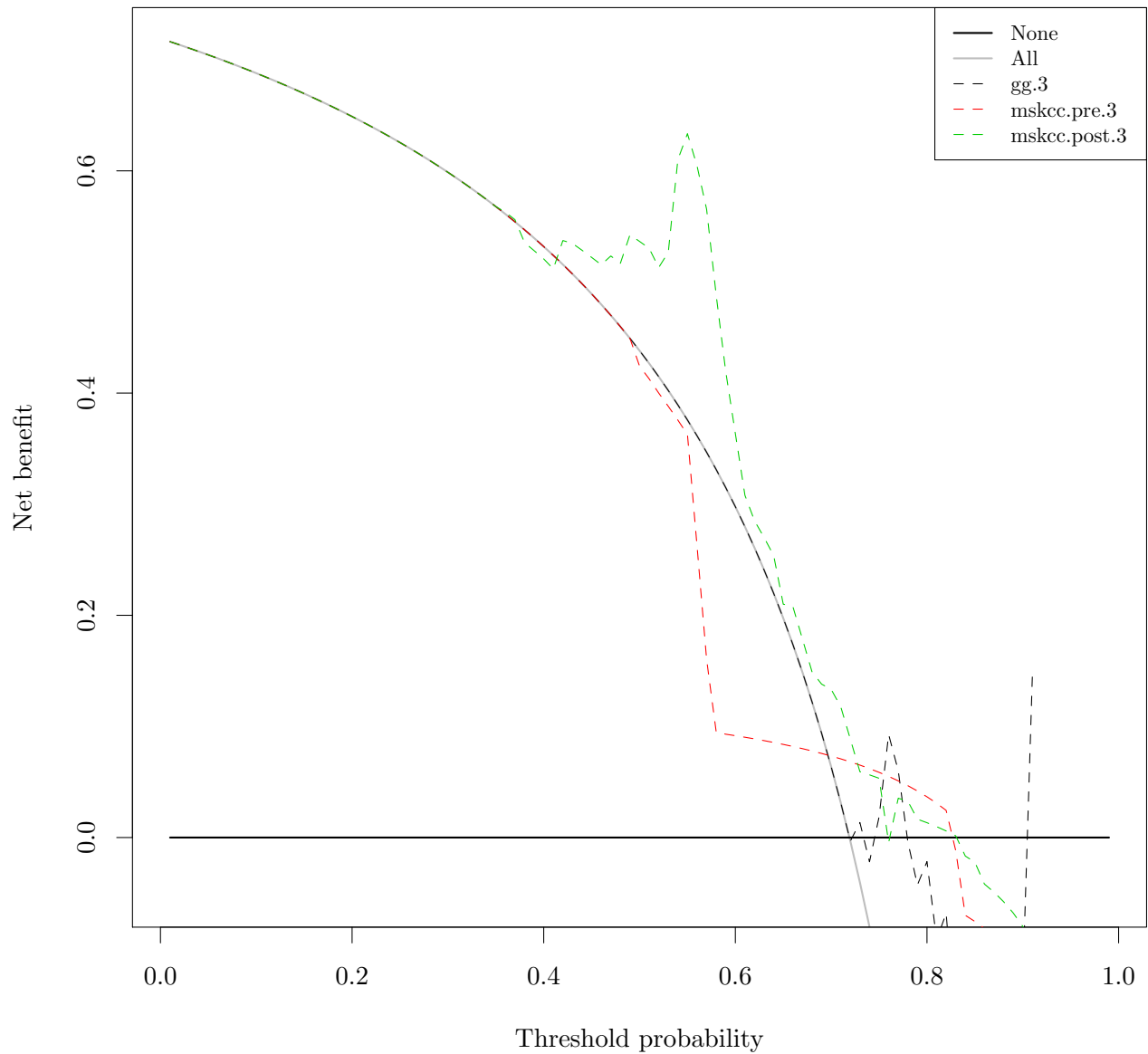
```
invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.2", "mskcc.pre.2", "mskcc.post.2")))

## [1] "gg.2: No observations with risk greater than 89% that have followup through the timepoint selected"
## [2] "mskcc.pre.2: No observations with risk greater than 85%, and therefore net benefit not calculable"
## [3] "mskcc.post.2: No observations with risk greater than 80% that have followup through the timepoint selected"
```



```
invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.3", "mskcc.pre.3", "mskcc.post.3")))

## [1] "gg.3: No observations with risk greater than 92% that have followup through the timepoint selected"
## [2] "mskcc.pre.3: No observations with risk greater than 95%, and therefore net benefit not calculable"
## [3] "mskcc.post.3: No observations with risk greater than 92% that have followup through the timepoint selected"
```



## 4.6 Brier score

```
calcIBS = function(surv, pred, pred_times, max_time, min_time = 0)
{
  stopifnot(nrow(surv) == nrow(pred) && length(pred_times) == ncol(pred))

  n = nrow(surv)
  marg_survfit = survfit(surv ~ 1)
  marg_censfit = survfit(Surv(surv[,1], !surv[,2]) ~ 1)
  marg_surv_func = approxfun(marg_survfit$time, marg_survfit$surv, method = "constant", yleft = 1, yright = 0)
  marg_cens_func = approxfun(marg_censfit$time, marg_censfit$surv, method = "constant", yleft = 1, yright = 0)

  pred_funcs = apply(pred, 1, function(pat_preds) approxfun(pred_times, pat_preds, yleft = 1, yright = 0))

  indiv_patient_bsc = function(pat_i, tstars)
  {
```

```

        observed_time = surv[pat_i, 1]
        observed_event = surv[pat_i, 2]
        pred_func = pred_funcs[[pat_i]]
        category = 1*(observed_time <= tstars & observed_event) + 2*(observed_time > tstars) + 3*(observed_time > tstars & !observed_event)
        bsc = rep(NA, length(tstars))
        bsc[category == 1] = pred_func(tstars[category == 1])^2 / marg_cens_func(observed_time[tstars[category == 1]])
        bsc[category == 2] = (1 - pred_func(tstars[category == 2]))^2 / marg_cens_func(tstars[category == 2])
        bsc[category == 3] = 0
        bsc
    }

    bsc_func = function(tstars) { rowMeans(sapply(1:n, function(pat_i) indiv_patient_bsc(pat_i, tstars))) }

    weight_func = function(tstars) { (1 - marg_surv_func(tstars)) / (1 - marg_surv_func(max_time)) }

    # Be slack and do trapezoidal int. with a fine grid. It should be possible
    # to calculate the int. exactly but I cbfed.
    int_grid = seq(min_time, max_time, length.out = 1e3)
    bsc_vals = bsc_func(int_grid)
    weight_vals = weight_func(int_grid)
    int_vals = bsc_vals * weight_vals
    ibsc = (2*sum(int_vals) - int_vals[1] - int_vals[length(int_vals)]) * (diff(range(int_grid))) / (max_time - min_time)

    return(list(bsc = bsc_vals, weights = weight_vals, eval_times = int_grid, ibsc = ibsc))
}

calcBSsingle = function(surv, pred, pred_time)
{
    n = nrow(surv)
    obs_time = surv[,1]
    obs_event = surv[,2]
    marg_censfit = survfit(Surv(obs_time, !obs_event) ~ 1)
    marg_cens_func = approxfun(marg_censfit$time, marg_censfit$surv, method = "constant", yleft = 1, yright = 0)

    brier_val = rep(NA, n)
    cat = 1*I(obs_time <= pred_time & obs_event) + 2*I(obs_time > pred_time) + 3*I(obs_time <= pred_time & !obs_event)
    brier_val[cat == 1] = (pred[cat == 1])^2 / marg_cens_func(obs_time[cat == 1])
    brier_val[cat == 2] = (1-pred[cat == 2])^2 / marg_cens_func(pred_time)
    brier_val[cat == 3] = 0

    mean(brier_val)
}

```

```

mskcc_post.12mo.glasgow.brier = calcBSsingle(Surv(data.glasgow$Time, data.glasgow$DSD), mskcc_post.12mo.glasgow.brier)
mskcc_post.24mo.glasgow.brier = calcBSsingle(Surv(data.glasgow$Time, data.glasgow$DSD), mskcc_post.24mo.glasgow.brier)
mskcc_post.36mo.glasgow.brier = calcBSsingle(Surv(data.glasgow$Time, data.glasgow$DSD), mskcc_post.36mo.glasgow.brier)
mskcc_pre.12mo.glasgow.brier = calcBSsingle(Surv(data.glasgow$Time, data.glasgow$DSD), mskcc_pre.12mo.glasgow.brier)
mskcc_pre.24mo.glasgow.brier = calcBSsingle(Surv(data.glasgow$Time, data.glasgow$DSD), mskcc_pre.24mo.glasgow.brier)
mskcc_pre.36mo.glasgow.brier = calcBSsingle(Surv(data.glasgow$Time, data.glasgow$DSD), mskcc_pre.36mo.glasgow.brier)
gg.path.glasgow.brier = calcIBS(Surv(data.glasgow$Time, data.glasgow$DSD), t(sapply(gg.path.glasgow, function(x) calcBSsingle(Surv(x$Time, x$DSD), x$brier))))
km0.path.glasgow.brier = calcIBS(Surv(data.glasgow$Time, data.glasgow$DSD), matrix(fit.km0$surv, nrow = nrow(fit.km0), byrow = TRUE))

```

```

mskcc_post.12mo.apgi.brier = calcBSsingle(Surv(data.apgi$Time, data.apgi$DSD), mskcc_post.12mo.apgi, 12/12)
mskcc_post.24mo.apgi.brier = calcBSsingle(Surv(data.apgi$Time, data.apgi$DSD), mskcc_post.24mo.apgi, 24/24)
mskcc_post.36mo.apgi.brier = calcBSsingle(Surv(data.apgi$Time, data.apgi$DSD), mskcc_post.36mo.apgi, 36/36)
mskcc_pre.12mo.apgi.brier = calcBSsingle(Surv(data.apgi$Time, data.apgi$DSD), mskcc_pre.12mo.apgi, 12/12)
mskcc_pre.24mo.apgi.brier = calcBSsingle(Surv(data.apgi$Time, data.apgi$DSD), mskcc_pre.24mo.apgi, 24/24)
mskcc_pre.36mo.apgi.brier = calcBSsingle(Surv(data.apgi$Time, data.apgi$DSD), mskcc_pre.36mo.apgi, 36/36)
gg.path.apgi.brier = calcIBS(Surv(data.apgi$Time, data.apgi$DSD), t(sapply(gg.path.apgi, function(x) x[,1:2])))
km0.path.apgi.brier = calcIBS(Surv(data.apgi$Time, data.apgi$DSD), matrix(fit.km0$surv, nrow = nrow(data.apgi), ncol = 2)))

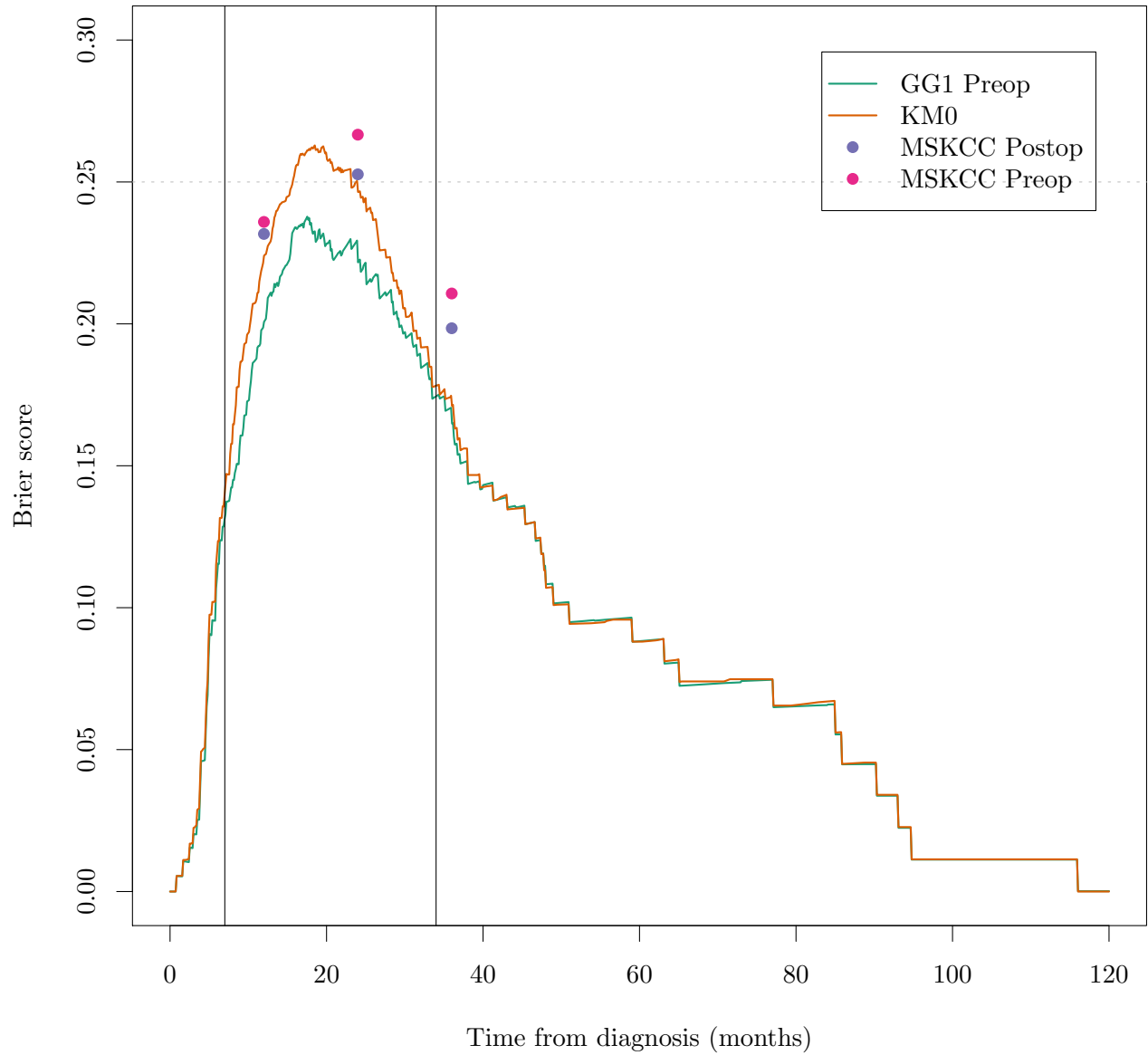
```

```

plot(gg.path.glasgow.brier$eval_times/365.25*12, gg.path.glasgow.brier$bsc, col = pal["gg"], type = "l", lwd = 2)
lines(gg.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc, col = pal["km0"], lwd = 2)
points(c(12, 24, 36), c(mskcc_post.12mo.glasgow.brier, mskcc_post.24mo.glasgow.brier, mskcc_post.36mo.glasgow.brier), col = pal["mskcc.post"], lty = "solid", lwd = 2)
points(c(12, 24, 36), c(mskcc_pre.12mo.glasgow.brier, mskcc_pre.24mo.glasgow.brier, mskcc_pre.36mo.glasgow.brier), col = pal["mskcc.pre"], lty = "solid", lwd = 2)
abline(h = 0.25, col = "grey", lty = "dotted")
abline(v = c(7, 34))
legend("topright",
      legend = c("GG1 Preop", "KM0", "MSKCC Postop", "MSKCC Preop"),
      pch = c(NA, NA, 16, 16),
      col = c(pal["gg"], pal["km0"], pal["mskcc.pre"], pal["mskcc.post"]),
      lty = c("solid", "solid", NA, NA),
      inset = 0.05, lwd = 2)

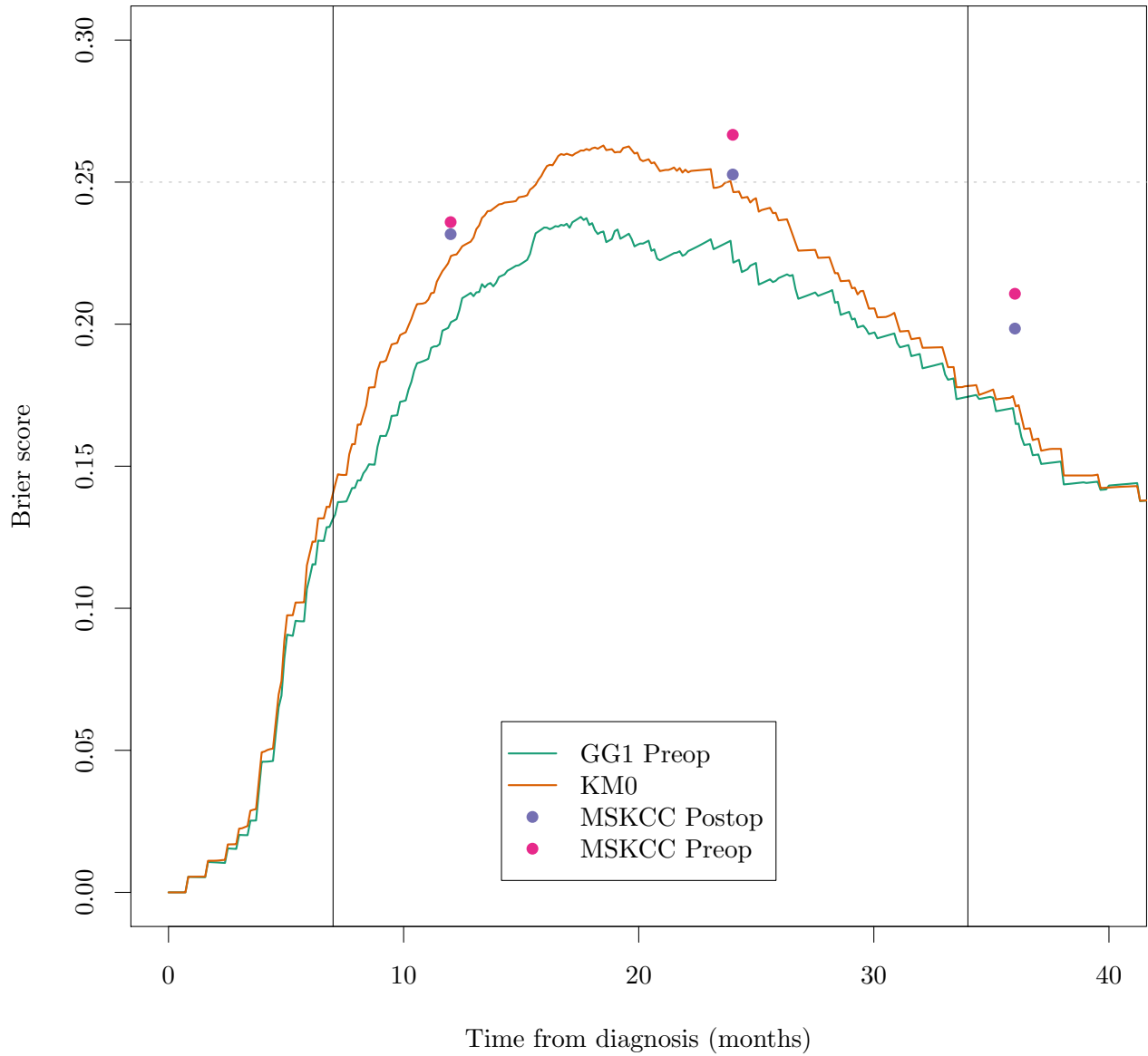
```

## Glasgow



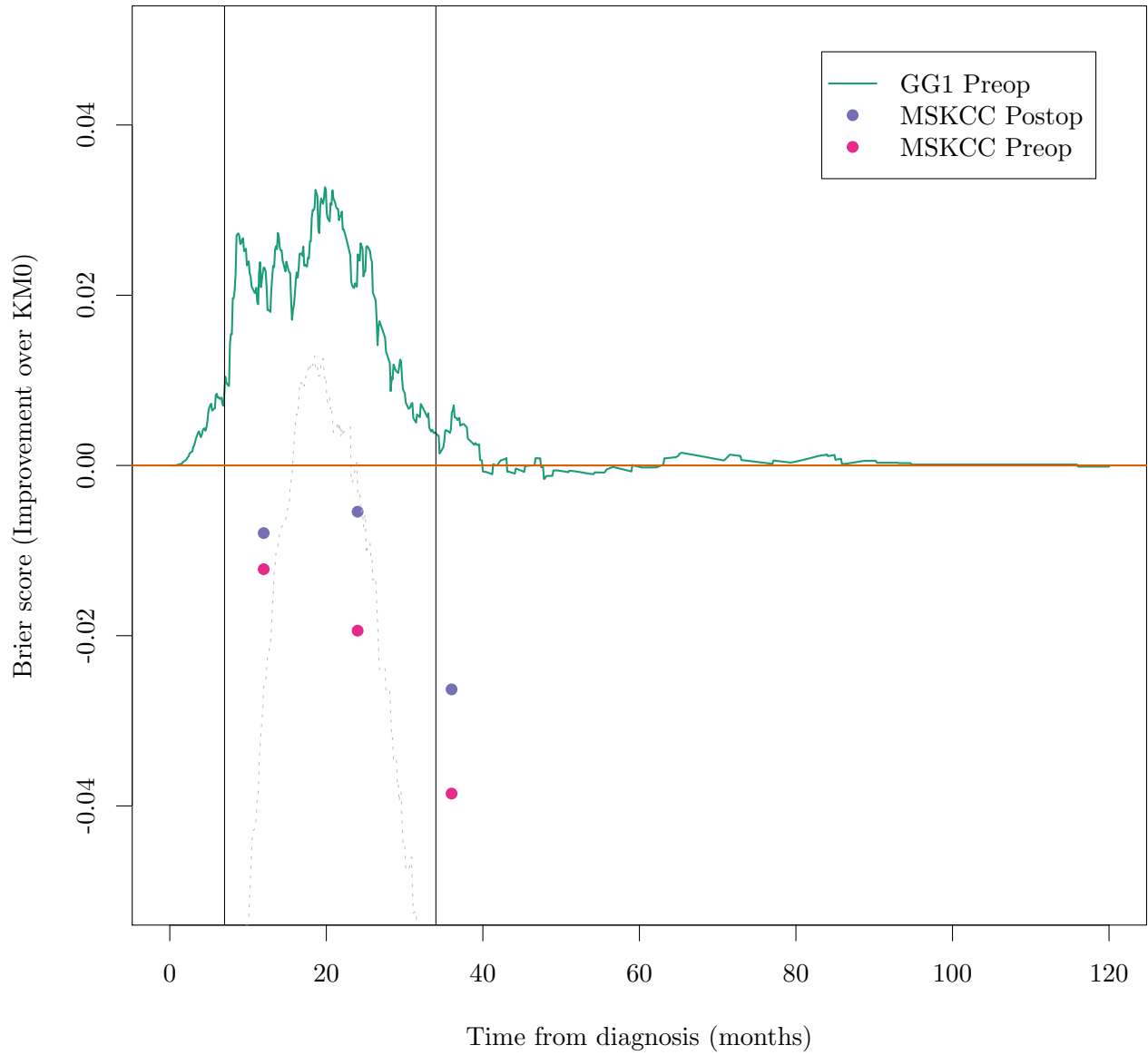
```
plot(gg.path.glasgow.brier$eval_times/365.25*12, gg.path.glasgow.brier$bsc, col = pal["gg"], type = "l",
lines(gg.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc, col = pal["km0"], lwd = 2),
points(c(12, 24, 36), c(mskcc_post.12mo.glasgow.brier, mskcc_post.24mo.glasgow.brier, mskcc_post.36mo.glasgow.brier), col = pal["mskcc.post"], lty = "n",
points(c(12, 24, 36), c(mskcc_pre.12mo.glasgow.brier, mskcc_pre.24mo.glasgow.brier, mskcc_pre.36mo.glasgow.brier), col = pal["mskcc.pre"], lty = "n",
abline(h = 0.25, col = "grey", lty = "dotted")
abline(v = c(7, 34))
legend("bottom",
      legend = c("GG1 Preop", "KM0", "MSKCC Postop", "MSKCC Preop"),
      pch = c(NA, NA, 16, 16),
      col = c(pal["gg"], pal["km0"], pal["mskcc.pre"], pal["mskcc.post"]),
      lty = c("solid", "solid", NA, NA),
      inset = 0.05, lwd = 2)
```

## Glasgow



```
plot(gg.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc - gg.path.glasgow.brier$bsc,
     points(c(12, 24, 36), approx(km0.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc, c(12, 24, 36),
     points(c(12, 24, 36), approx(km0.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc, c(12, 24, 36),
     lines(gg.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc - 0.25, col = "grey", lty = 1),
     abline(v = c(7, 34))
     abline(h = 0, col = pal["km0"], lwd = 2)
     legend("topright",
           legend = c("GG1 Preop", "MSKCC Postop", "MSKCC Preop"),
           pch = c(NA, 16, 16),
           col = c(pal["gg"], pal["mskcc.pre"], pal["mskcc.post"]),
           lty = c("solid", NA, NA),
           inset = 0.05, lwd = 2)
```

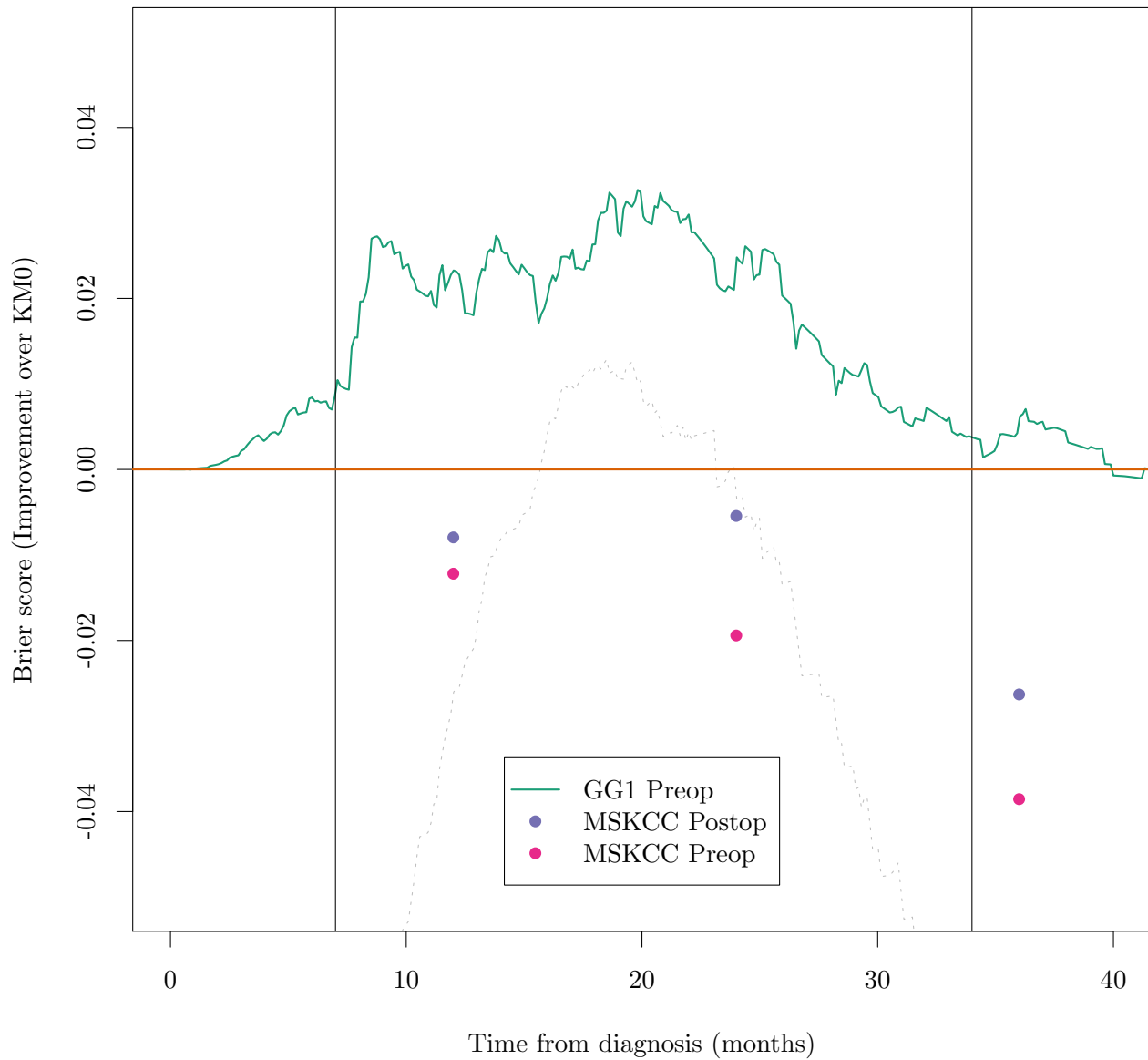
## Glasgow



```
plot(gg.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc - gg.path.glasgow.brier$bsc,
points(c(12, 24, 36), approx(km0.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc, c(
points(c(12, 24, 36), approx(km0.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc, c(
lines(gg.path.glasgow.brier$eval_times/365.25*12, km0.path.glasgow.brier$bsc - 0.25, col = "grey", lty =
abline(v = c(7, 34))
abline(h = 0, col = pal["km0"], lwd = 2)
legend("bottom",
  legend = c("GG1 Preop", "MSKCC Postop", "MSKCC Preop"),
  pch = c(NA, 16, 16),
  col = c(pal["gg"], pal["mskcc.pre"], pal["mskcc.post"]),
  lty = c("solid", NA, NA),
  inset = 0.05, lwd = 2)
```

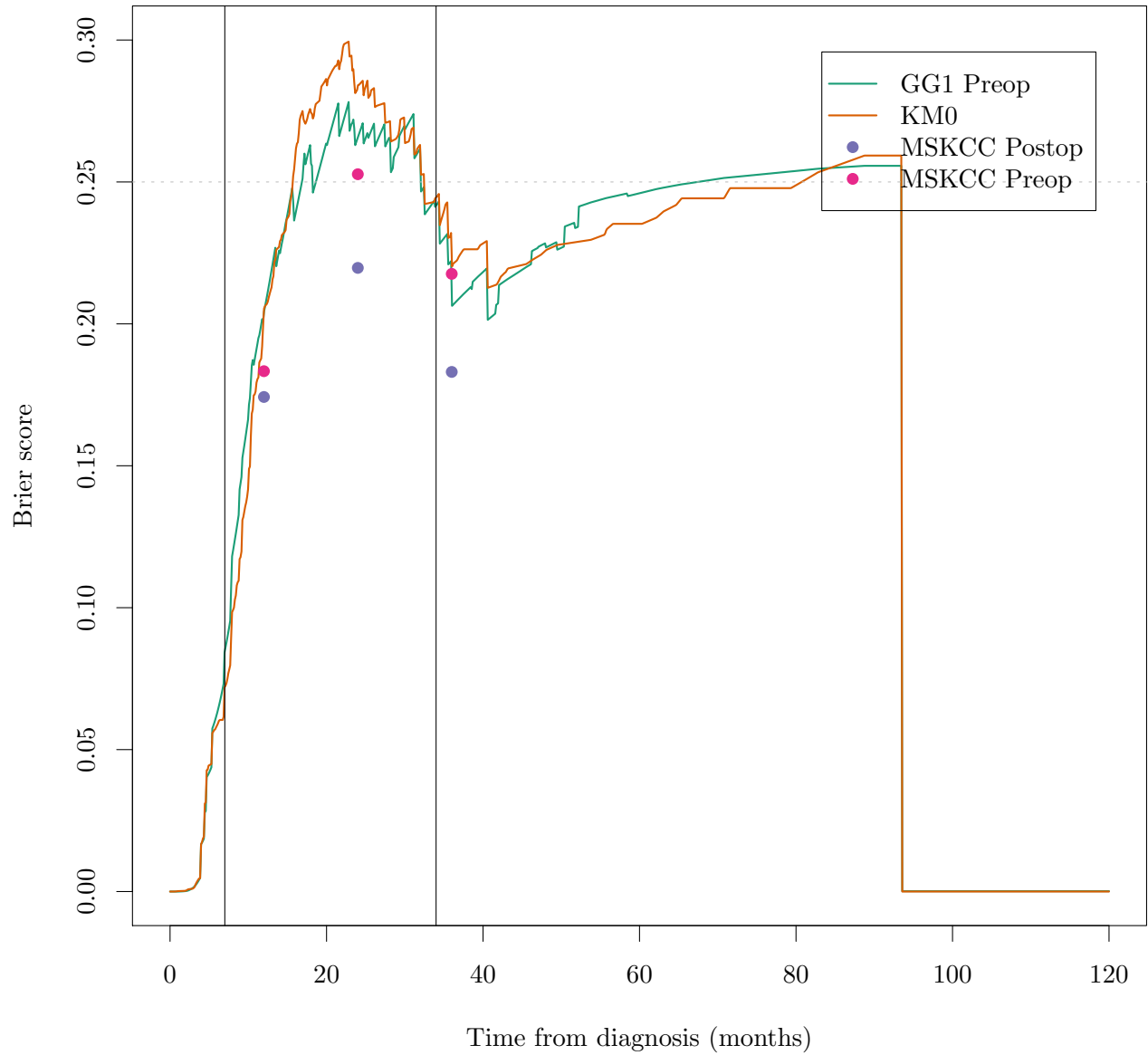


## Glasgow



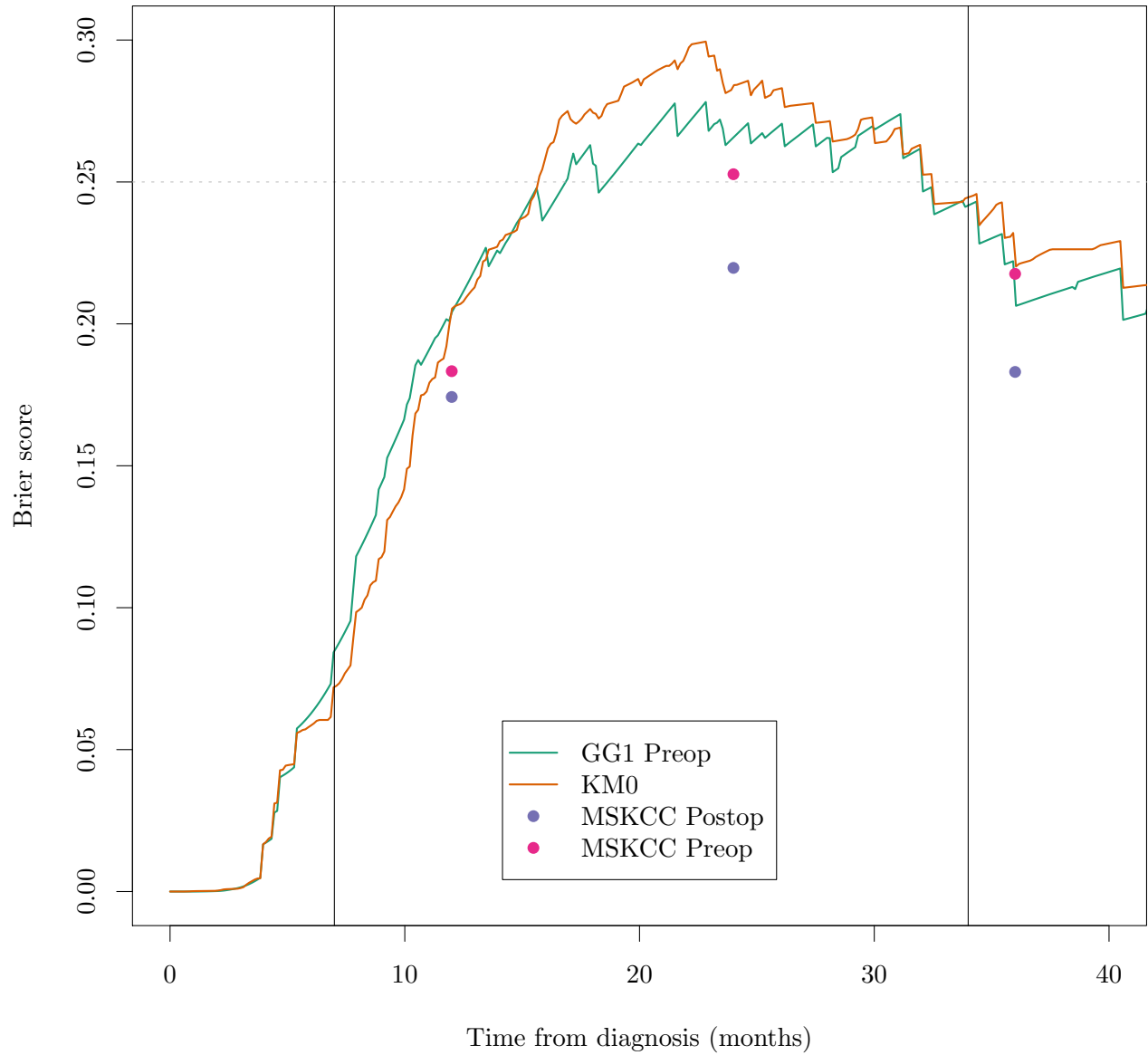
```
plot(gg.path.apgi.brier$eval_times/365.25*12, gg.path.apgi.brier$bsc, col = pal["gg"], type = "l", ylim = c(-0.04, 0.04))
lines(gg.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc, col = pal["km0"], lwd = 2)
points(c(12, 24, 36), c(mskcc_post.12mo.apgi.brier, mskcc_post.24mo.apgi.brier, mskcc_post.36mo.apgi.brier), col = pal["mskcc.post"], pch = 16)
points(c(12, 24, 36), c(mskcc_pre.12mo.apgi.brier, mskcc_pre.24mo.apgi.brier, mskcc_pre.36mo.apgi.brier), col = pal["mskcc.pre"], pch = 16)
abline(h = 0.25, col = "grey", lty = "dotted")
abline(v = c(7, 34))
legend("topright",
      legend = c("GG1 Preop", "KM0", "MSKCC Postop", "MSKCC Preop"),
      pch = c(NA, NA, 16, 16),
      col = c(pal["gg"], pal["km0"], pal["mskcc.pre"], pal["mskcc.post"]),
      lty = c("solid", "solid", NA, NA),
      inset = 0.05, lwd = 2)
```

## APGI



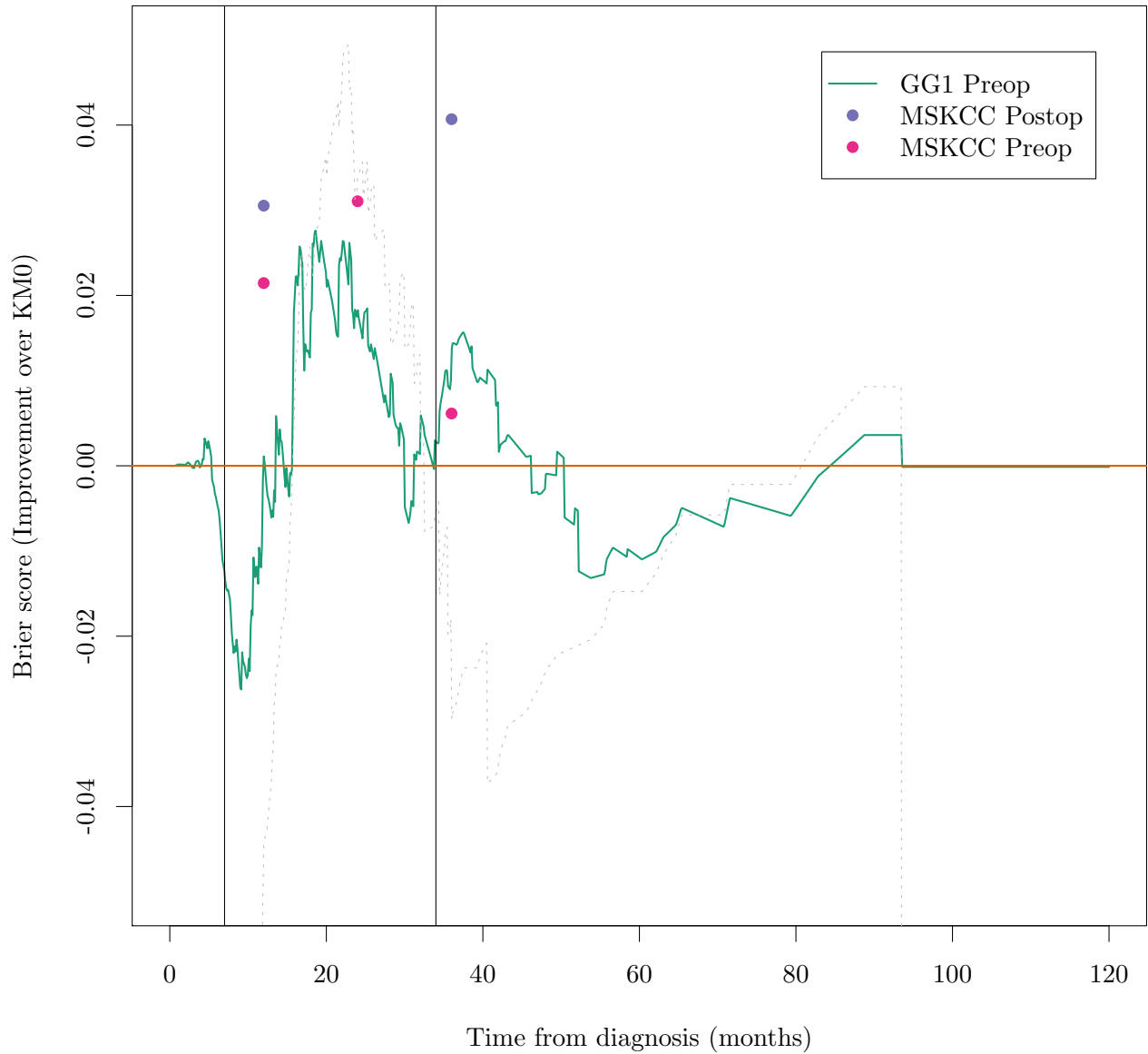
```
plot(gg.path.apgi.brier$eval_times/365.25*12, gg.path.apgi.brier$bsc, col = pal["gg"], type = "l", ylim = c(0, 0.3))
lines(gg.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc, col = pal["km0"], lwd = 2)
points(c(12, 24, 36), c(mskcc_post.12mo.apgi.brier, mskcc_post.24mo.apgi.brier, mskcc_post.36mo.apgi.brier), col = pal["mskcc.post"], pch = 16)
points(c(12, 24, 36), c(mskcc_pre.12mo.apgi.brier, mskcc_pre.24mo.apgi.brier, mskcc_pre.36mo.apgi.brier), col = pal["mskcc.pre"], pch = 16)
abline(h = 0.25, col = "grey", lty = "dotted")
abline(v = c(7, 34))
legend("bottom",
      legend = c("GG1 Preop", "KM0", "MSKCC Postop", "MSKCC Preop"),
      pch = c(NA, NA, 16, 16),
      col = c(pal["gg"], pal["km0"], pal["mskcc.pre"], pal["mskcc.post"]),
      lty = c("solid", "solid", NA, NA),
      inset = 0.05, lwd = 2)
```

## APGI



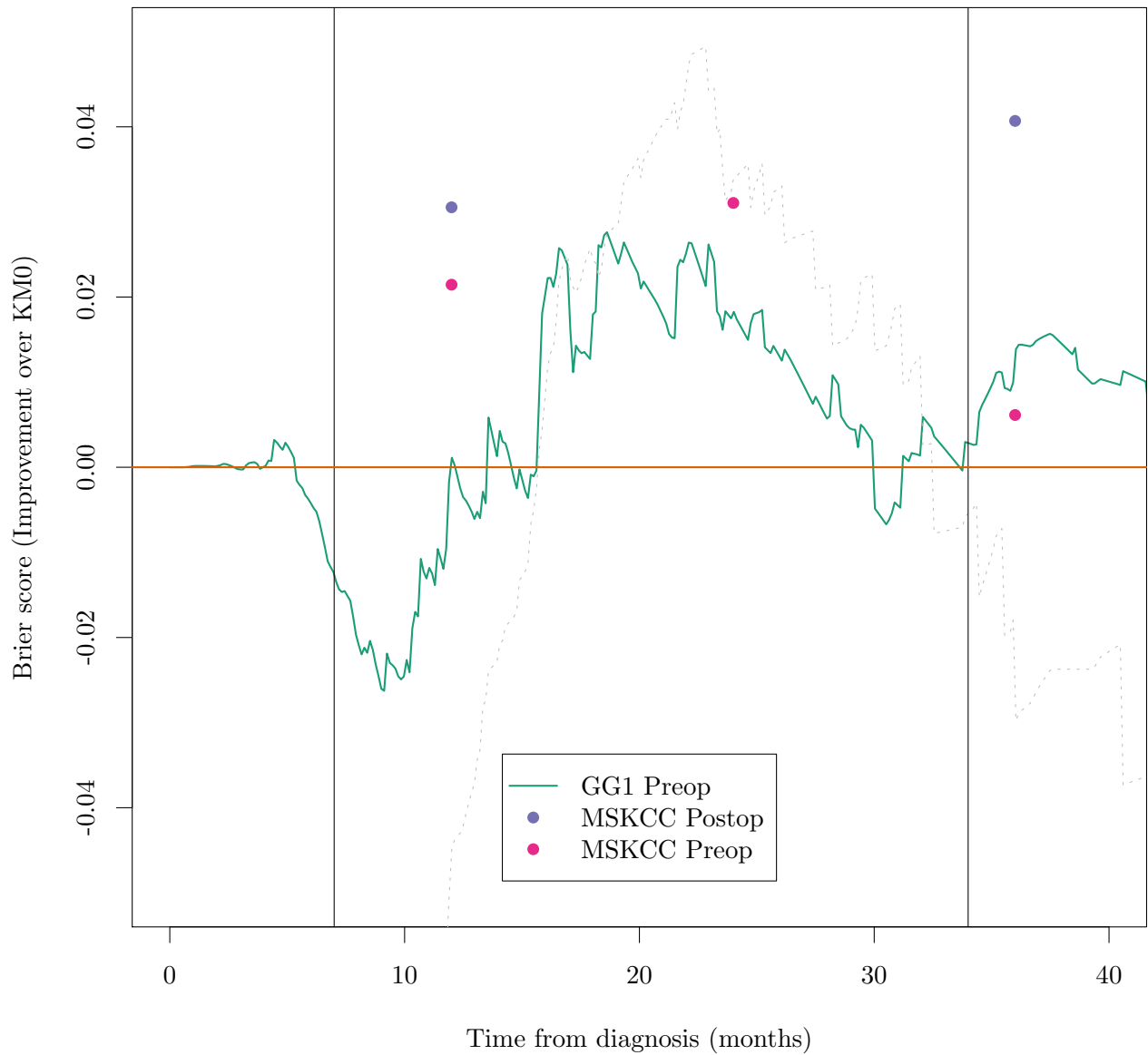
```
plot(gg.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc - gg.path.apgi.brier$bsc, col = "grey", lty = "dotted",
     points(c(12, 24, 36), approx(km0.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc, c(12, 24, 36)),
           points(c(12, 24, 36), approx(km0.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc, c(12, 24, 36))),
     lines(gg.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc - 0.25, col = "grey", lty = "dotted"),
     abline(v = c(7, 34))
     abline(h = 0, col = pal["km0"], lwd = 2)
     legend("topright",
           legend = c("GG1 Preop", "MSKCC Postop", "MSKCC Preop"),
           pch = c(NA, 16, 16),
           col = c(pal["gg"], pal["mskcc.pre"], pal["mskcc.post"]),
           lty = c("solid", NA, NA),
           inset = 0.05, lwd = 2)
```

## APGI



```
plot(gg.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc - gg.path.apgi.brier$bsc, col = pal["km0"], lwd = 2)
points(c(12, 24, 36), approx(km0.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc, c(12, 24, 36)), col = "grey", lty = "dotted")
points(c(12, 24, 36), approx(km0.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc, c(12, 24, 36)), col = "grey", lty = "dotted")
lines(gg.path.apgi.brier$eval_times/365.25*12, km0.path.apgi.brier$bsc - 0.25, col = "green", lty = "solid")
abline(v = c(7, 34))
abline(h = 0, col = pal["km0"], lwd = 2)
legend("bottom",
      legend = c("GG1 Preop", "MSKCC Postop", "MSKCC Preop"),
      pch = c(NA, 16, 16),
      col = c(pal["gg"], pal["mskcc.pre"], pal["mskcc.post"]),
      lty = c("solid", NA, NA),
      inset = 0.05, lwd = 2)
```

## APGI



```
probs_bs_boot_func_glasgow = function(d, i) {
  bs.mskcc.postop.12 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_post.12mo.glasgow[i], 12/12*365.25)
  bs.mskcc.postop.24 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_post.24mo.glasgow[i], 24/12*365.25)
  bs.mskcc.postop.36 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_post.36mo.glasgow[i], 36/12*365.25)
  bs.mskcc.preop.12 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_pre.12mo.glasgow[i], 12/12*365.25)
  bs.mskcc.preop.24 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_pre.24mo.glasgow[i], 24/12*365.25)
  bs.mskcc.preop.36 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_pre.36mo.glasgow[i], 36/12*365.25)

  bs.gg.vals = t(sapply(gg.path.glasgow[i], function(path) approx(path[,1], path[,2], c(12, 24, 36))))
  rownames(bs.gg.vals) <- NULL
  bs.gg.12 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), bs.gg.vals[,1], 12/12*365.25)
  bs.gg.24 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), bs.gg.vals[,2], 24/12*365.25)
  bs.gg.36 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), bs.gg.vals[,3], 36/12*365.25)
}
```

```

bs.km0.vals = approx(fit.km0$time, fit.km0$surv, c(12, 24, 36)/12*365.25)$y
bs.km0.12 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), rep(bs.km0.vals[1], nrow(d[i,])), 12/12*365.25)
bs.km0.24 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), rep(bs.km0.vals[2], nrow(d[i,])), 24/12*365.25)
bs.km0.36 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), rep(bs.km0.vals[3], nrow(d[i,])), 36/12*365.25)

result = c(
  bs.gg.12 - bs.km0.12,          bs.mskcc.preop.12 - bs.km0.12,
  bs.gg.12 - bs.mskcc.preop.12,
  bs.gg.24 - bs.km0.24,          bs.mskcc.preop.24 - bs.km0.24,
  bs.gg.24 - bs.mskcc.preop.24,
  bs.gg.36 - bs.km0.36,          bs.mskcc.preop.36 - bs.km0.36,
  bs.gg.36 - bs.mskcc.preop.36)

names(result) <- NULL
result
}

set.seed(20150208)
deltaBrier.boot.glasgow = boot(data.glasgow, probs_bs_boot_func_glasgow, R = 500)
deltaBrier.boot.glasgow.cis = t(sapply(1:ncol(deltaBrier.boot.glasgow$t), function(i) boot.ci(deltaBrier.boot.glasgow, i, R = 500)))
colnames(deltaBrier.boot.glasgow.cis) = c("level", "lowindex", "highindex", "lci", "uci")
rownames(deltaBrier.boot.glasgow.cis) = c(
  "12:gg-km0", "12:pre-km0", "12:gg-pre",
  "24:gg-km0", "24:pre-km0", "24:gg-pre",
  "36:gg-km0", "36:pre-km0", "36:gg-pre")
deltaBrier.boot.glasgow

##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = data.glasgow, statistic = probs_bs_boot_func_glasgow,
##       R = 500)
##
##
## Bootstrap Statistics :
##      original      bias      std. error
## t1* -0.023252 -5.591e-04   0.011020
## t2*  0.012000  5.097e-04   0.014791
## t3* -0.035252 -1.069e-03   0.018703
## t4* -0.024707 -1.173e-03   0.011163
## t5*  0.020378  1.780e-04   0.020822
## t6* -0.045085 -1.351e-03   0.022651
## t7* -0.006137 -3.073e-04   0.006092
## t8*  0.039775 -9.123e-06   0.018277
## t9* -0.045912 -2.982e-04   0.018448

deltaBrier.boot.glasgow.cis

##           level lowindex highindex      lci      uci
## 12:gg-km0   0.95    19.36    493.3 -0.0438016 0.0001641
## 12:pre-km0   0.95    10.07    485.4 -0.0179132 0.0401415
## 12:gg-pre    0.95     9.88    485.4 -0.0753277 -0.0035136
## 24:gg-km0   0.95    17.35    492.2 -0.0471870 -0.0023731

```

```
## 24:pre-km0  0.95    11.87    487.8 -0.0189747  0.0617515
## 24:gg-pre   0.95    19.24    493.3 -0.0845755  0.0024417
## 36:gg-km0   0.95    15.48    490.9 -0.0174246  0.0056702
## 36:pre-km0  0.95     7.75    482.0  0.0002576  0.0703455
## 36:gg-pre   0.95    17.88    492.7 -0.0791661 -0.0078058
```

```
probs_bs_boot_func_apgi = function(d, i) {
  bs.mskcc.postop.12 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_post.12mo.apgi[i], 12/12*365.25)
  bs.mskcc.postop.24 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_post.24mo.apgi[i], 24/12*365.25)
  bs.mskcc.postop.36 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_post.36mo.apgi[i], 36/12*365.25)
  bs.mskcc.preop.12 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_pre.12mo.apgi[i], 12/12*365.25)
  bs.mskcc.preop.24 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_pre.24mo.apgi[i], 24/12*365.25)
  bs.mskcc.preop.36 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), mskcc_pre.36mo.apgi[i], 36/12*365.25)

  bs.gg.vals = t(sapply(gg.path.apgi[i], function(path) approx(path[,1], path[,2], c(12, 24, 36)/12*365.25)$y
rownames(bs.gg.vals) <- NULL
  bs.gg.12 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), bs.gg.vals[,1], 12/12*365.25)
  bs.gg.24 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), bs.gg.vals[,2], 24/12*365.25)
  bs.gg.36 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), bs.gg.vals[,3], 36/12*365.25)

  bs.km0.vals = approx(fit.km0$time, fit.km0$surv, c(12, 24, 36)/12*365.25)$y
  bs.km0.12 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), rep(bs.km0.vals[1], nrow(d[i,])), 12/12*365.25)
  bs.km0.24 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), rep(bs.km0.vals[2], nrow(d[i,])), 24/12*365.25)
  bs.km0.36 = calcBSsingle(Surv(d$Time[i], d$DSD[i]), rep(bs.km0.vals[3], nrow(d[i,])), 36/12*365.25)

  result = c(
    bs.gg.12 - bs.km0.12,          bs.mskcc.preop.12 - bs.km0.12,
    bs.gg.12 - bs.mskcc.preop.12,
    bs.gg.24 - bs.km0.24,          bs.mskcc.preop.24 - bs.km0.24,
    bs.gg.24 - bs.mskcc.preop.24,
    bs.gg.36 - bs.km0.36,          bs.mskcc.preop.36 - bs.km0.36,
    bs.gg.36 - bs.mskcc.preop.36)

  names(result) <- NULL
  result
}

set.seed(20150208)
deltaBrier.boot.apgi = boot(data.apgi, probs_bs_boot_func_apgi, R = 500)
deltaBrier.boot.apgi.cis = t(sapply(1:ncol(deltaBrier.boot.apgi$t), function(i) boot.ci(deltaBrier.boot.apgi, i, R = 500)$ci[,2:3]))
colnames(deltaBrier.boot.apgi.cis) = c("level", "lowindex", "highindex", "lci", "uci")
rownames(deltaBrier.boot.apgi.cis) = c(
  "12:gg-km0", "12:pre-km0", "12:gg-pre",
  "24:gg-km0", "24:pre-km0", "24:gg-pre",
  "36:gg-km0", "36:pre-km0", "36:gg-pre")
deltaBrier.boot.apgi

##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = data.apgi, statistic = probs_bs_boot_func_apgi, R = 500)
```

```
##
##
## Bootstrap Statistics :
##      original      bias    std. error
## t1* -0.00113 -0.0011691    0.015701
## t2* -0.02190 -0.0009299    0.018710
## t3*  0.02077 -0.0002392    0.028129
## t4* -0.01807  0.0005315    0.013458
## t5* -0.03102 -0.0029566    0.030885
## t6*  0.01295  0.0034881    0.034386
## t7* -0.01368  0.0004382    0.008461
## t8* -0.00230 -0.0019783    0.031044
## t9* -0.01138  0.0024165    0.031451
```

```
deltaBrier.boot.apgi.cis
```

```
##           level lowindex highindex      lci      uci
## 12:gg-km0   0.95    14.99    490.7 -0.02914 0.029597
## 12:pre-km0  0.95    19.61    493.7 -0.05458 0.021352
## 12:gg-pre   0.95    11.55    487.5 -0.03594 0.073835
## 24:gg-km0   0.95    14.16    490.0 -0.04298 0.010080
## 24:pre-km0  0.95    24.25    495.0 -0.08215 0.036547
## 24:gg-pre   0.95     6.77    478.8 -0.06156 0.073555
## 36:gg-km0   0.95     6.89    480.5 -0.03200 0.001481
## 36:pre-km0  0.95    13.87    489.6 -0.06168 0.053566
## 36:gg-pre   0.95    15.08    490.8 -0.06278 0.051051
```

```
temp.time = gsub(".*", "", rownames(deltaBrier.boot.glasgow.cis))
temp.methodpos = gsub(".*", "", gsub("-", "", rownames(deltaBrier.boot.glasgow.cis)))
temp.methodneg = gsub(".*-", "", rownames(deltaBrier.boot.glasgow.cis))
temp.methods = sort(unique(c(temp.methodpos, temp.methodneg)))
tapply(1:length(temp.time), temp.time, function(is) {
  res = matrix(0, nrow = length(temp.methods), ncol = length(temp.methods))
  rownames(res) = temp.methods
  colnames(res) = temp.methods
  # Make res signed. 0 => NS. +1 => row is better than col (BS_row - BS_col < 0). -1 => row is
  res[cbind(temp.methodpos[is], temp.methodneg[is])] = (sign(deltaBrier.boot.glasgow.cis[is, "uci"]
  res[cbind(temp.methodneg[is], temp.methodpos[is])] = (sign(deltaBrier.boot.glasgow.cis[is, "uci"]
  res
})

## $`12`
##      gg km0 pre
## gg    0  0  1
## km0   0  0  0
## pre  -1  0  0
##
## $`24`
##      gg km0 pre
## gg    0  1  0
## km0  -1  0  0
## pre   0  0  0
##
## $`36`
```



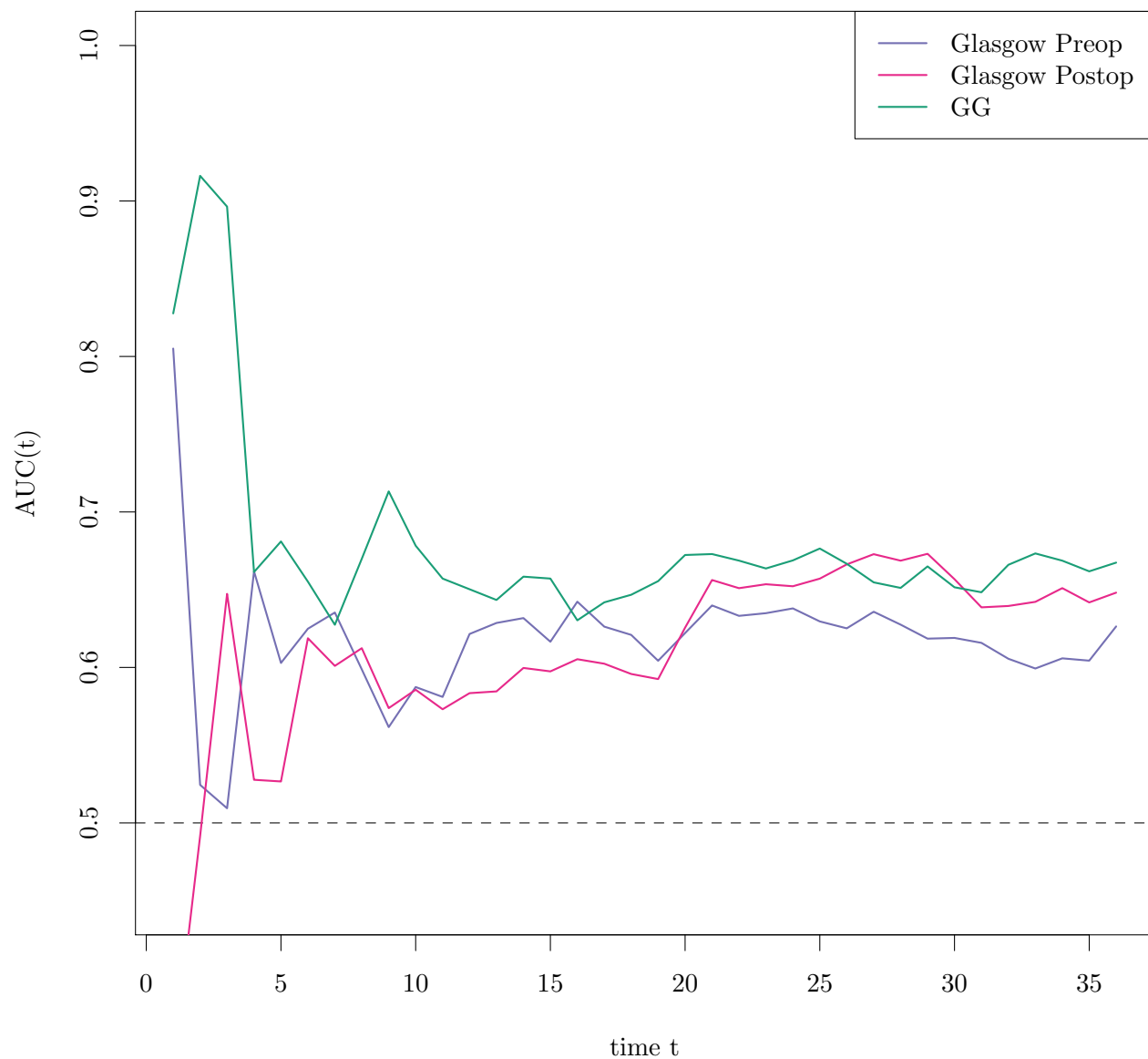
```
##      gg km0 pre
## gg    0  0  1
## km0   0  0  1
## pre  -1 -1  0
```

```
temp.time = gsub(".*", "", rownames(deltaBrier.boot.apgi.cis))
temp.methodpos = gsub(".*:", "", gsub(".*", "", rownames(deltaBrier.boot.apgi.cis)))
temp.methodneg = gsub(".*-", "", rownames(deltaBrier.boot.apgi.cis))
temp.methods = sort(unique(c(temp.methodpos, temp.methodneg)))
tapply(1:length(temp.time), temp.time, function(is) {
  res = matrix(0, nrow = length(temp.methods), ncol = length(temp.methods))
  rownames(res) = temp.methods
  colnames(res) = temp.methods
  # Make res signed. 0 => NS. +1 => row is better than col (BS_row - BS_col < 0). -1 => row is
  res[cbind(temp.methodpos[is], temp.methodneg[is])] = (sign(deltaBrier.boot.apgi.cis[is, "uci"]))
  res[cbind(temp.methodneg[is], temp.methodpos[is])] = (sign(deltaBrier.boot.apgi.cis[is, "uci"]))
  res
})

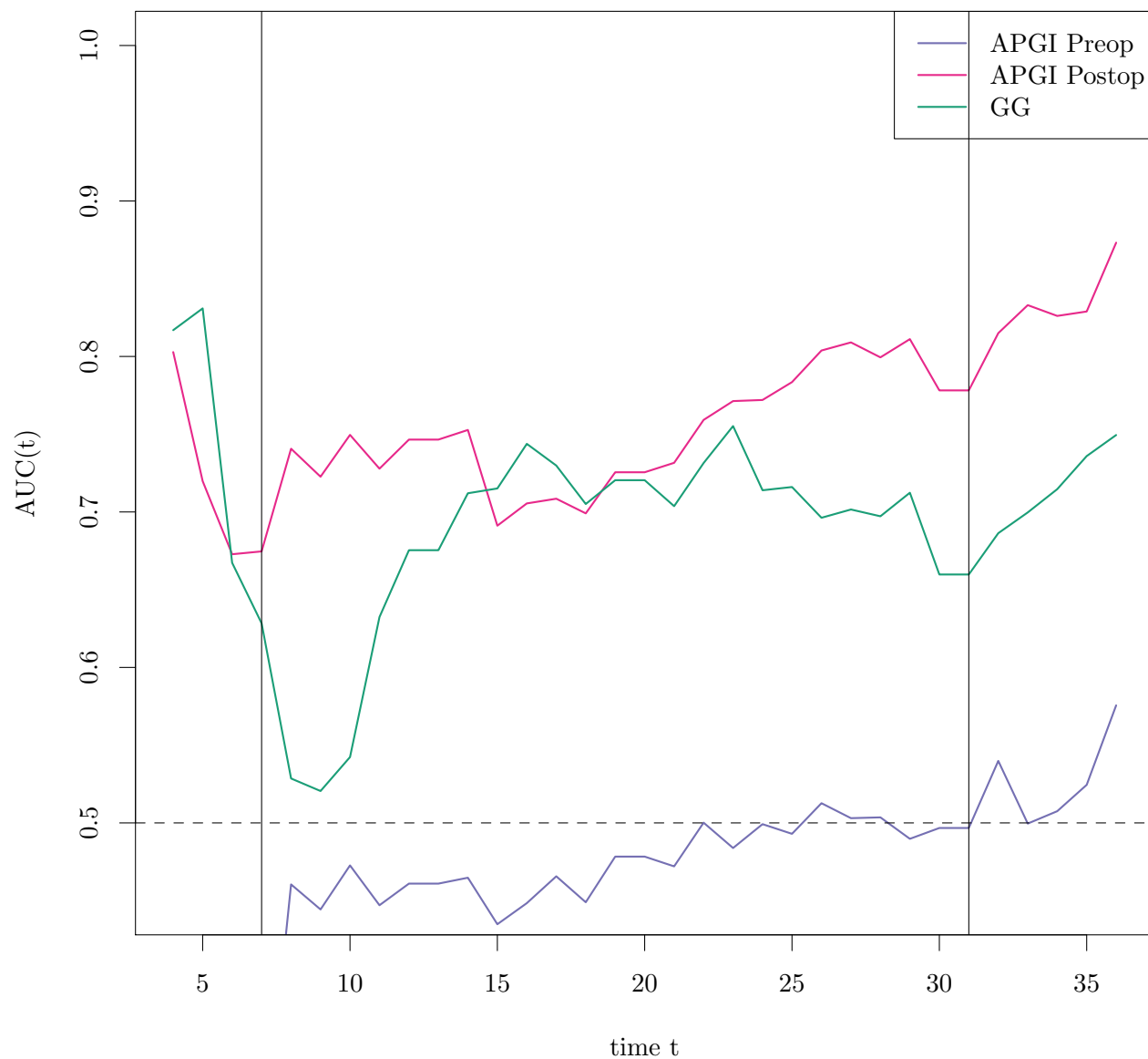
## $`12`
##      gg km0 pre
## gg    0  0  0
## km0   0  0  0
## pre   0  0  0
##
## $`24`
##      gg km0 pre
## gg    0  0  0
## km0   0  0  0
## pre   0  0  0
##
## $`36`
##      gg km0 pre
## gg    0  0  0
## km0   0  0  0
## pre   0  0  0
```

Cumulative-dynamic:

```
mskcc_pre.cdroc.glasgow = timeROC(data.glasgow$Time/365.25*12, data.glasgow$DSD, mskcc_pre.linpred.glasgow)
mskcc_post.cdroc.glasgow = timeROC(data.glasgow$Time/365.25*12, data.glasgow$DSD, mskcc_post.linpred.glasgow)
gg.cdroc.glasgow = timeROC(data.glasgow$Time/365.25*12, data.glasgow$DSD, gg.linpred.glasgow, cause = 1)
plotAUCcurve(mskcc_pre.cdroc.glasgow, conf.int = FALSE, add = FALSE, col = pal["mskcc.pre"])
plotAUCcurve(mskcc_post.cdroc.glasgow, conf.int = FALSE, add = TRUE, col = pal["mskcc.post"])
plotAUCcurve(gg.cdroc.glasgow, conf.int = FALSE, add = TRUE, col = pal["gg"])
legend("topright", legend = c("Glasgow Preop", "Glasgow Postop", "GG"), col = c(pal["mskcc.pre"], pal["mskcc.post"], pal["gg"])
```

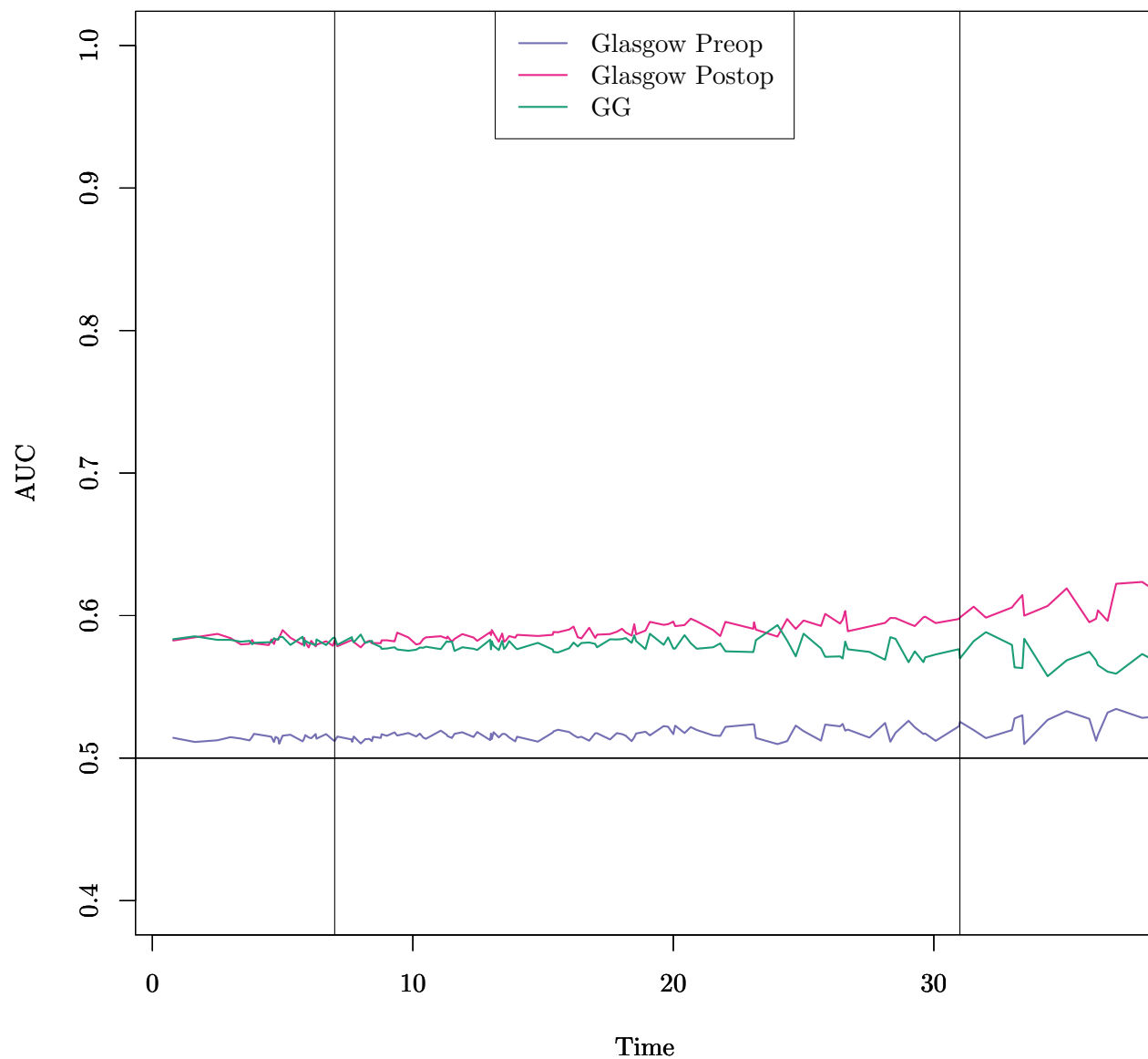


```
mskcc_pre.cdroc.apgi = timeROC(data.apgi$Time/365.25*12, data.apgi$DSD, mskcc_pre.linpred.apgi, cause =
mskcc_post.cdroc.apgi = timeROC(data.apgi$Time/365.25*12, data.apgi$DSD, mskcc_post.linpred.apgi, cause =
gg.cdroc.apgi = timeROC(data.apgi$Time/365.25*12, data.apgi$DSD, gg.linpred.apgi, cause = 1, times = se
plotAUCcurve(mskcc_pre.cdroc.apgi, conf.int = FALSE, add = FALSE, col = pal["mskcc.pre"])
plotAUCcurve(mskcc_post.cdroc.apgi, conf.int = FALSE, add = TRUE, col = pal["mskcc.post"])
plotAUCcurve(gg.cdroc.apgi, conf.int = FALSE, add = TRUE, col = pal["gg"])
legend("topright", legend = c("APGI Preop", "APGI Postop", "GG"), col = c(pal["mskcc.pre"], pal["mskcc.p
abline(v = c(7, 31))
```



Incident-dynamic:

```
invisible(risksetAUC(data.glasgow$Time/365.25*12, status = data.glasgow$DSD, marker = mskcc_pre.linpred,
par(new = TRUE)
invisible(risksetAUC(data.glasgow$Time/365.25*12, status = data.glasgow$DSD, marker = mskcc_post.linpred,
par(new = TRUE)
invisible(risksetAUC(data.glasgow$Time/365.25*12, status = data.glasgow$DSD, marker = gg.linpred.glasgow,
par(new = TRUE)
legend("top", legend = c("Glasgow Preop", "Glasgow Postop", "GG"), col = c(pal["mskcc.pre"], pal["mskcc.post"], pal["gg"]),
abline(v = c(7, 31))
```



```
invisible(risksetAUC(data.apgi$Time/365.25*12, status = data.apgi$DSD, marker = mskcc_pre.linpred.apgi,
par(new = TRUE)
invisible(risksetAUC(data.apgi$Time/365.25*12, status = data.apgi$DSD, marker = mskcc_post.linpred.apgi,
par(new = TRUE)
invisible(risksetAUC(data.apgi$Time/365.25*12, status = data.apgi$DSD, marker = gg.linpred.apgi, tmax =
par(new = TRUE)
legend("top", legend = c("APGI Preop", "APGI Postop", "GG"), col = c(pal["mskcc.pre"], pal["mskcc.post"],
abline(v = c(7, 31))
```

