Creating Nomograms with SGplot

Cindy Loman

Genomic Health, Inc.

cloman@genomichealth.com

Introduction

Physician decision-making often involves consideration of multiple factors in estimating the risk of a clinical outcome such as relapse, recovery, treatment response or treatment failure. In medical research, multivariate modeling with logistic regression (PROC LOGISTIC) is often used to build prediction models that incorporate multiple factors. Nomograms are a way to translate a complex model into a user-friendly tool that a physician can use in his/her clinical practice.

Background

The concept of a nomogram was introduced by Philbert Maurice d'Ocagne in 1884 as an engineering tool for the pre-calculator era. Historically in the medical field, they have played a role in estimating the correct drug dosages a patient should receive. Today, if you search online for medical nomograms you could find many developed by various prominent cancer treatment centers. However, these are not graphs at all, but dashboards where you enter clinical information and computer code returns a probablility.

Objectives

The objective in this poster is to return the nomogram to its graphical origins, where the value of each covariate in predicting outcome is readily apparent. This will primarily be accomplished using proc logistic and proc sgplot.

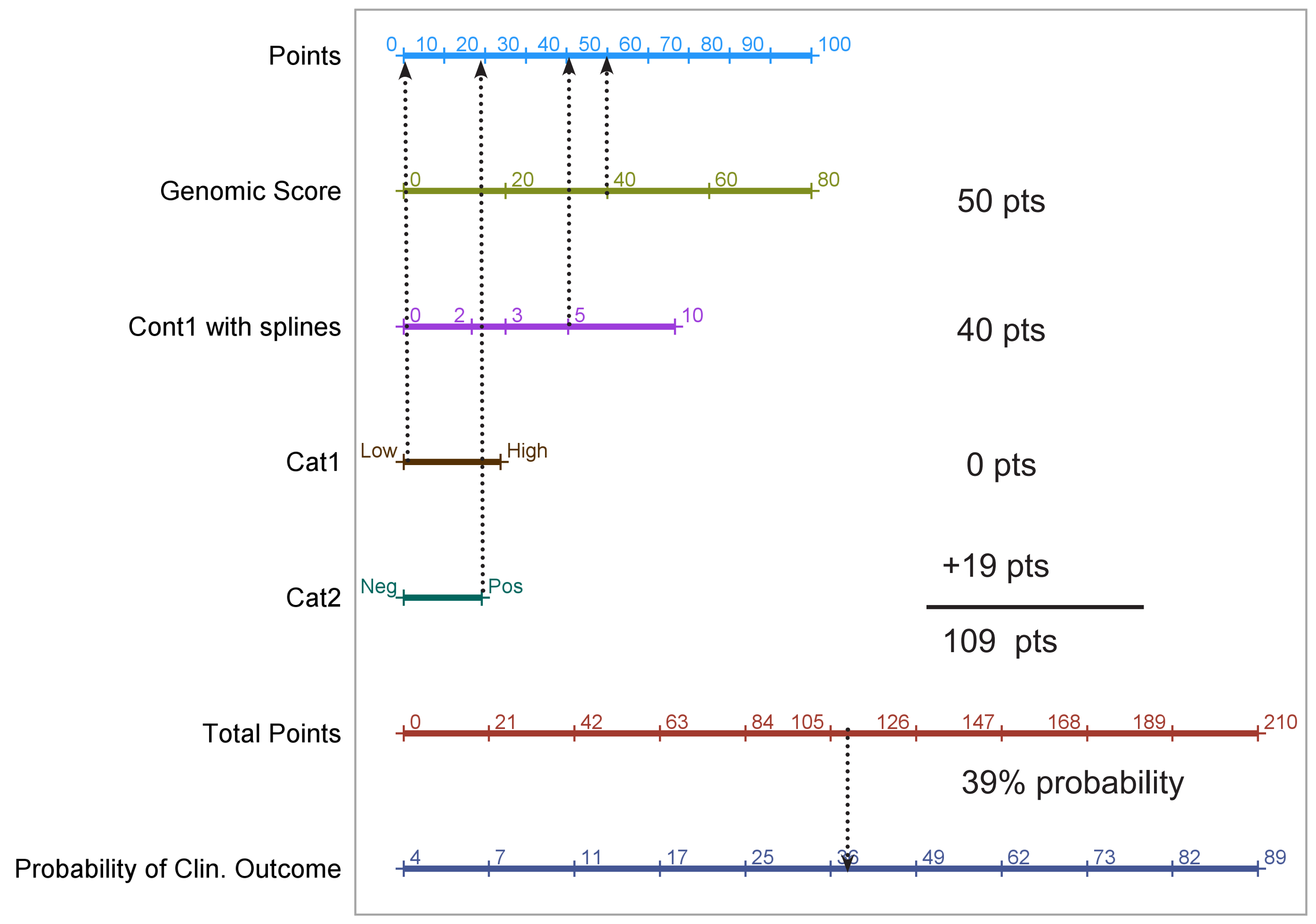
How does the nomogram work?

For the purposes of illustration, a hypothetical scenario is used in which a continuous genomic score (ranging in value from 0 to 80) is combined with 3 traditional risk factors:

- Continuous, ranging from 0 to 10
- Categorical, with values of High and Low (coded as 1 and 0, resp.)
- Categorical, with values of Positive and Negative (coded as 1 and 0, resp.)

to predict a clinical outcome. The nomogram is depicted below, along with a description of how it is used.

Nomogram for the Prediction of Clinical Outcome



Suppose that a patient's genomic score is 40, their cont var =5, their cat1 var = Low, and their cat2 var = Positive?

The physician would:

1. Use a straight edge to draw lines from each of the covariates of interest to the Points scale at the top of the nomogram.
2. Compute the sum of the 4 Points values (50 pts for Genomic Score, 40 pts for the continuous covariate, 0 pts for the first categorical variable, and 19 pts for the second categorical variable)
3. Use a straight edge to convert the Total Points to the Probability of Clinical Outcome at the bottom of the nomogram.

Data preparation

Manipulation of the data in preparation for graphing is much more complicated than the graphing itself. The categorical variables for Cat1 and Cat2 were recoded to have values of 0 or 1. For the continuous variable, there will be 2 variables in the model, Cont_Basis1 and Cont_Basis2, which need to be combined when reporting out for Cont.

Genomic Score	Cat1	Cat2	Cont_basis1	Cont_basis2	Order	Order_fmt
0	0	0	0	0	6	Genomic Score
20	0	0	0	0	6	Genomic Score
40	0	0	0	0	6	Genomic Score
60	0	0	0	0	6	Genomic Score
80	0	0	0	0	6	Genomic Score
0	0	0	0	0	4	Cat1
0	1	0	0	0	4	Cat1
0	0	0	0	0	3	Cat2
0	0	1	0	0	3	Cat2
0	0	0	0	0	5	Cont1 with Splines
0	0	0	2	-0.165	5	Cont1 with Splines
0	0	0	3	-3.131	5	Cont1 with Splines
0	0	0	5	-35.662	5	Cont1 with Splines
0	0	0	10	-395.056	5	Cont1 with Splines

The results from logistic analysis of the original data set were used to score a dataset that included a range of values for each covariate while setting the others to 0. Each line on the graph corresponds to the order variable and each row in this dataset has an analogous marker symbol on the graph.

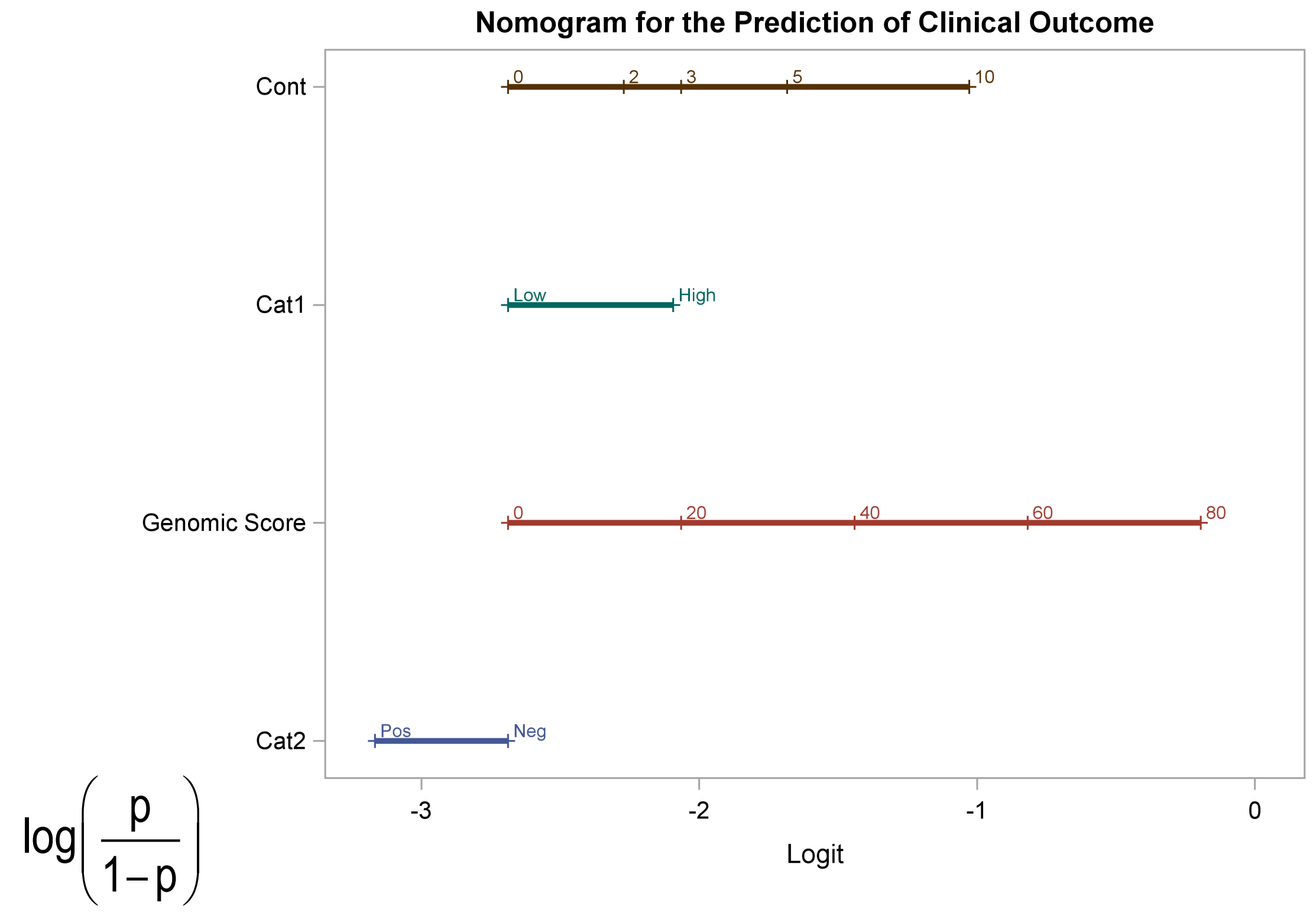
Working with splines

In order to create the natural spline with 2 df for the cont variable, the following macro was conveniently inserted into the data step code. Before doing that, however, it was necessary to create the values for the knots using proc univariate.

```
proc univariate data=indsn;
     var cont;
     output out = percentiles pctlpts = 1,50,99 pctlpre=cont_;
run;

/****not showing the addition of the percentiles to the data******/

%macro natspline2(var=,knot1=,knot2=,knot3=,basis1=,basis2=);
     qqlambda2 = (&knot3. - &knot2.) / (&knot3. - &knot1.);
     &basis1 = &var;
     &basis2 = max(0,(&var.-&knot2.)**3) -
     qqlambda2*max(0,(&var.-&knot1.)**3)-
     (1-qqlambda2)*max(0,(&var.-&knot3.)**3);

     drop qqlambda2;
%mend natspline2;
```

First version

Nomogram for the Prediction of Clinical Outcome



Problems

- Cat2 is going the wrong direction.
- Logit scale is not user friendly.
- Probability not calculated.

Solutions

- Reverse the 0/1 coding for Cat2
- Remove the x-axis labels and ticks.
- Add a points scale at the top.
- Add a cumulative points scale at end.
- Repeat the cumulative points scale but label with the associated p-values.

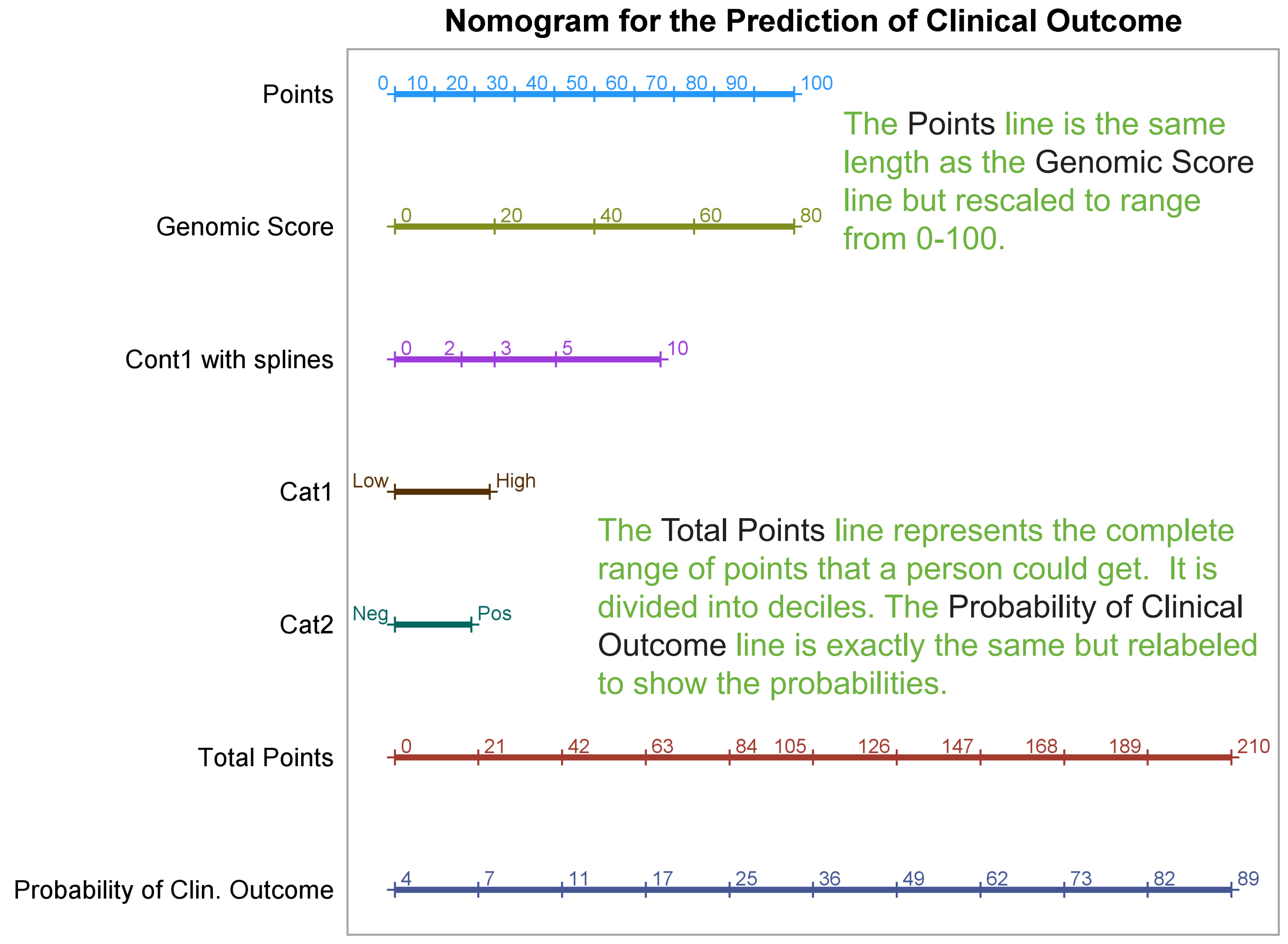
Calculating the Probability

On the graph, the role of the intercept in the logistic model is ignored when calculating the line length for each covariate. However, it is used when determining the probability of clinical outcome. In the code snippet below the EST array contains the betas for each variable.

```
data bars;
   set covdata;
   array vars [5] GeneScore cat1 cat2 cont_basis1 cont_basis2 ;
   array est [5] est_GeneScore est_cat1 est_cat2 est_cont_basis1
                 est_cont_basis2 ;
   array bar [5] bar_GeneScore bar_cat1 bar_cat2 bar_cont_basis1
                 bar_cont_basis2;
   model=est_intercept;
   do i=1 to 5; *calculate the model result;
      bar[i]= vars[i]*est[i];*calculate the value of each variable;
      model=model + bar[i];
   end;
   prob=1/(exp(-1*(model)) +1); *keep intercept for probability;
   barlength=model-est_intercept; *remove intercept for bar length;
run;
```

Final version

Nomogram for the Prediction of Clinical Outcome



The Points line is the same length as the Genomic Score line but rescaled to range from 0-100.

The Total Points line represents the complete range of points that a person could get. It is divided into deciles. The Probability of Clinical Outcome line is exactly the same but relabeled to show the probabilities.

Graphics code

When it comes to coding this graph, it is fairly straightforward. Unlike most graphs, though, the tick marks and tick values are not shown on the x-axis.

```
proc sgplot data=points noautolegend;
     xaxis display=(nolabel noticks novalues)
           offsetmin=.05 offsetmax=.05;
     yaxis display=(nolabel noticks) values=(1 to 7 by 1)
           offsetmin=.05   offsetmax=.05
           nteger tickvalueformat=ordf. ;
     reg x=barlength y=order/ group=order datalabel=mylabels
                         lineattrs=(pattern=1 thickness=3)
                          markerattrs=(symbol="plus");
run;
```

Conclusions

- SGPLOT can be used to create user-friendly prediction nomograms from multivariable logistic regression models that demonstrate the relative importance of each predictor.
- Such nomograms may have particular utility in settings where online access to modern dashboards is limited (e.g., in the developing world).