

fullnen-report

L3P3

Wednesday, November 19, 2014

This document summarizes the second approach in order to expand the elastic net regularized predictor to include node awareness. This third iteration substitutes the event type field with the concatenation of the node and the event type *only for the prediction column*, effectively allowing us to apply the same models to this new data adding a layer of node awareness. The whole process of model creation is described in the file “nen3-whole.R”, found in the private L3P3 Github repository. The models are exported in a variable called “models_sc”, which is a list of all the combinations of nodes and events in the system, containing each one the best created model for each event.

```
require(ggplot2)
```

```
## Loading required package: ggplot2
```

```
load('data.Rdata')
```

Our objective is now to study the performance of this models. To do so, we’re going to create a matrix where the rows are the system nodes and the columns are the events that happened on the system. The coefficients of the matrix will be the f-score of each model in each node.

Let’s now extract some insightful analytics.

- First, let’s see how many models were correctly created and how many models failed to create:

```
## [1] "Possible models:" "1335"
```

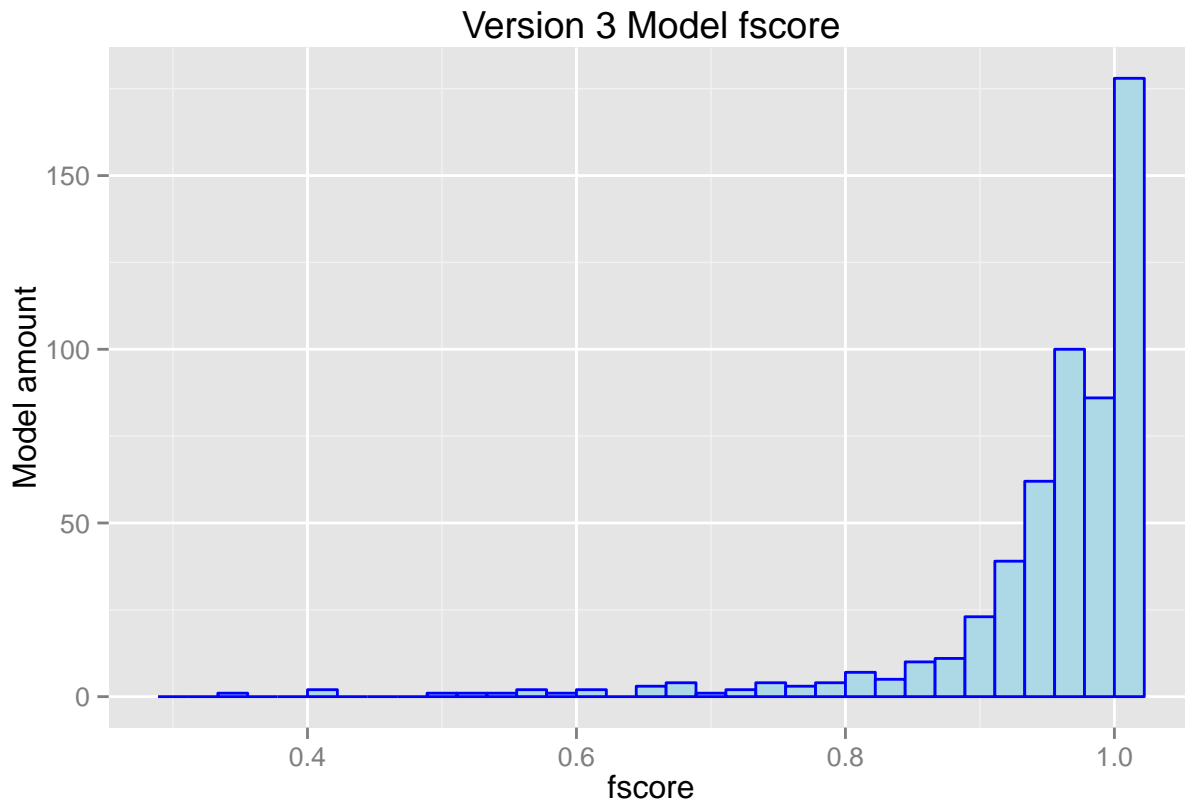
```
## [1] "Created models:" "553"
```

```
## [1] "Failed models:" "782"
```

The percentage of created models is a 42%. A logical result, as each model has now far fewer data to train.

- Now, we’ll study obtained fscores in the 563 correctly created models:

```
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
```



The histogram shows good results for the models that actually work. A 86.7992767% (480 models) is at least 0.9 and a 94.2133816% (521 models) is at least 0.8. The average obtained fscore of the created models is 0.9465066, a slightly lower amount compared to the one found in the previous iteration. These models are hugely simpler than those, though, which makes still a feasible option in deployment.

- Lastly, we will study which critical events were correctly modelled and their fscore. There are three different critical events on the system, which we will study separately:
 - 96731154: this event indicates a critical threshold violation. It only appears in two nodes.

```
## fwpeuib02.sceu.corp fwpeuid02.sceu.corp
##                                0                                1
```

* 68917: this event's message points that a device has stopped responding to external requests and/or p

```
##      aswpeuib02.sceu.corp      aswpeuib04.sceu.corp
##      0.0000000              0.0000000
##      aswpeuid01.sceu.corp      aswpeuid02.sceu.corp
##      0.0000000              0.0000000
##      aswpeuid03.sceu.corp      aswpeuid04.sceu.corp
##      0.9767442              1.0000000
##      aswpeuin01.sceu.corp      aswpeuin02.sceu.corp
##      0.0000000              0.0000000
##      aswpeuin03.sceu.corp      aswpeuin06.sceu.corp
##      0.0000000              0.9777778
##      aswpeuin08.sceu.corp      aswpeuin09.sceu.corp
```

##	1.0000000	0.0000000
##	aswpeuin10.sceu.corp	aswpeuin12.sceu.corp
##	0.9600000	0.0000000
##	ASWPEUIN13.sceu.corp	ASWPEUIN19.sceu.corp
##	0.0000000	1.0000000
##	dnsrespeuin02-adm.sceu.corp	dswpeuin01.sceu.corp
##	0.0000000	0.0000000
##	dswpeuin02.sceu.corp	dswpeuin03.sceu.corp
##	0.0000000	0.0000000
##	dswpeuin04.sceu.corp	dswpeuwt01.sceu.corp
##	0.0000000	0.0000000
##	dswpeuwt02.sceu.corp	dswpeuwt03.sceu.corp
##	0.0000000	0.0000000
##	dswpeuwt04.sceu.corp	fwpeuid02.sceu.corp
##	1.0000000	0.0000000

A total of 26 models are created, out of which 7 were correctly created, this is, a 26.9230769%, with an average fscore of 0.9877889 and a standard deviation of 0.0081416.

* 69481:this event's message poitns that a whole chassis has stopped responding to polls. While it is g

...

##	aswpeuib02.sceu.corp	aswpeuib04.sceu.corp	aswpeuid01.sceu.corp
##	0.0000000	0.0000000	0.0000000
##	aswpeuid02.sceu.corp	aswpeuid03.sceu.corp	aswpeuid04.sceu.corp
##	0.9677419	1.0000000	0.0000000
##	aswpeuin01.sceu.corp	aswpeuin02.sceu.corp	aswpeuin03.sceu.corp
##	0.0000000	0.0000000	1.0000000
##	aswpeuin06.sceu.corp	aswpeuin08.sceu.corp	aswpeuin10.sceu.corp
##	0.0000000	0.0000000	1.0000000
##	aswpeuin12.sceu.corp	ASWPEUIN13.sceu.corp	ASWPEUIN19.sceu.corp
##	0.9600000	0.0000000	1.0000000
##	dswpeuin01.sceu.corp	dswpeuin02.sceu.corp	dswpeuin03.sceu.corp
##	0.0000000	1.0000000	0.0000000
##	dswpeuin04.sceu.corp	dswpeuwt01.sceu.corp	dswpeuwt02.sceu.corp
##	0.0000000	0.9795918	0.9600000
##	dswpeuwt03.sceu.corp	dswpeuwt04.sceu.corp	
##	1.0000000	0.0000000	

...

A total of 23 models are created, out of which 10 were correctly created, this is, a 43.4782609%, with an average fscore of 0.9867334 and a standard deviation of 0.0089707.

The next step we will take will be to add resource consumption information to the models.