

Kepler - Bug #5611

LinearModel actor doesn't handle input variables properly

05/22/2012 02:50 PM - Jim Regetz

Status:	New	Start date:	05/22/2012
Priority:	Normal	Due date:	
Assignee:	Derik Barseghian	% Done:	0%
Category:	actors	Estimated time:	0.00 hour
Target version:	Unspecified	Spent time:	0.00 hour
Bugzilla-Id:	5611		

Description

The R code in the RExpression-based LinearModel actor has several problems:

1. In a linear model, it's the *independent* (predictor) variables that can be either numeric or factor (categorical). The dependent (response) variable must be numeric. The actor code gets this reversed, and fails with an error depending on the inputs.
2. The conditional `is.character()` in the final 'if' statement always evaluates to FALSE because of the conversion-to-factor in the first 'if' statement.
3. The intercept and slope aren't reported out in any useful way, and the intercept isn't even printed to the console output (if displayed).

I'd suggest the following replacement, which doesn't conform exactly to the intended behavior of the original actor, but I think provides a nice starter template for doing a univariate linear model fit. I also changed it to emit the fitted model object itself, which could be passed to another actor for summarizing, generating an ANOVA table, extracting coefficient estimates, etc.

Note that I wouldn't usually explicitly store the model formula as an object, but here I think it's useful to indicate the mapping of input ports to model variables right at the top.

```
#-----
```

```
model <- Dependent ~ Independent
```

```
if (is.character(Independent)) {  
  Independent <- factor(Independent)  
}
```

1. fit model; fitted lm object is available on output port
`results <- lm(model)`
`print(summary(results))`

1. plot data, adding regression line if appropriate
`plot(model)`
`if (is.numeric(Independent)) {`
 `abline(results, col="red")`
`}`

```
#-----
```

History

#1 - 03/27/2013 02:31 PM - Redmine Admin

Original Bugzilla ID was 5611