Searching for Dusty Corners: Understanding the Prediction of the Cross Section of Returns

Carlos Carvalho, John Cochrane, Juhani Linnainmaa, Rob McCulloch

1. Goals
2. Predictability
3. Fit-the-Fit, Where are the nonlinearities and interactions ??
4. Basic fit-the-fit without T
5. fit-fit with Time
6. Seeing the Nonlinearity, T in the tree, Rolling fits
7. Seeing the non-GAM part with T
8. Conclusion

1. Goals

Predict

We use simple approaches to predicting the monthly cross section of firm returns using variables obtained in the previous month.

We have 62 variables measured on a firm this month which attempt to provide useful predictive information about the return next month.

Focus on methods that could work well with little tuning:

- linear
- simple trees
- random forests
- BART

Our goal is not to get the ultimate return predictor.

Our goal is is to use relatively straighforward predictive methods that can capture nonlinearities and then interpret the results.

However,

Juhani has extensively studied portfolio formation based on out of sample BART based predictions
and reports strong performance !!!!

Interpret

fit the fit

We will see that predictions based on monthly BART fits are competitive.

To understand the BART predictions we fit a single tree where the response is the prediction and use the simpler tree structure to glean interpretable characteristics of the relationship between the predictions and the predictors.

See "Model Interpretation Through Lower-Dimensional Posterior Summarization",
Wood, Carvalho, Murray.

Dusty Corners:

We think there are small parts of the predictor space where interesting nonlinearities kick in.

We will try to indentify variables that contribute to nonlinearity and interactions in the dusty corners.

For example, a dusty corner corresponds the prediction of unusually small returns.

Interesting non-linearites kick in for extreme returns, not so much in the middle.

We just want simple ways of seeing this.

Data:

```
> length(dates)
[1] 1253753
> dates[1]
[1] 196306
> dates[1253753]
[1] 202005
```

- 684 months of data, June 1963 to May 2020.
- Each month we have a cross section of firm returns, and 62 firm characteristics measured in the previous month.
- 1,253,753 total observations on a firm return and vector of characteristics.
- threw out "tinies"
- on a monthly basis express each x as a quantile in $(0, 1)$.
- regression imputation of missing values
- monthly demean returns, so we are predicting amount above average

We can start predicting after the first 10 years of data.

```
> dim(ddf)
[1] 1114198      63

> names(ddf)
 [1] "accruals"      "assetgrowth"   "BE/ME"         "cashflowtoequ"
 [5] "aturnoverchg"  "debtissuance"  "earningsprice" "EM"
 [9] "grossprofitab" "inventorygrow" "herfindahl"    "netoperatinga"
[13] "piotroskiF"    "abnormalinves" "leverage"      "accruals_beme"
[17] "netwcap_chang" "oscore"        "profitmargin"  "profitability"
[21] "returnonequit" "salesgrowth"   "salestoprice"  "sustainablegr"
[25] "total_xfin"    "zscore"        "indadjcapxgro" "salesminusinv"
[29] "investmenttoc" "invgrowthrate" "I/A"           "qmj_profitabi"
[33] "chs_distress"  "ffprofitabili" "organizationc" "advertising"
[37] "opleverage"    "rd"            "tax"           "Profitability"
[41] "dssdur"        "disttohigh"    "amihud"        "marketbeta"
[45] "firmage"       "Volatility"    "ME"            "LT_reversals"
[49] "maxdailyret"   "r_12,2"        "intmomentum"   "Price"
[53] "seasonality"   "seasonality_o" "ST_reversals"  "Volume"
[57] "divmonth"      "sharevolume"   "coskewness"    "divyield"
[61] "momreversal"   "BE/ME*"        "R"
```

1,114,198 observations after the first 10 years = 120 months of data.

Some Key Predictor Variables

For example, these variables turn out to be particularly interesing:

ST_reversals:
prior one month return. "short term reversals".

r_12,2:
prior one year return, skipping a month. "momentum effect".

Volatility

disttohigh
distance to high

Panel B: List of return predictors

Category	Predictor	Category	Predictor
Investment,	Asset growth	Price-scaled	Book-to-market
growth, and	Inventory growth	predictors	Monthly book-to-market
duration	Sales growth		Cash-flow to equity
	Sustainable growth		Enterprise multiple
	Ind.-adjusted CAPX growth		Sales to price
	Growth in sales-inventory		
	Investment to capital	Financing	Total external financing
	Investment growth rate	and	Dividend month
	Investment to assets	payouts	Dividend yield
	Abnormal investment		Debt issuance
	Equity duration		
		Expenses	Advertising
Operational	Accruals		R&D
efficiency,	Change in asset turnover		Taxes
earnings	Accruals and book-to-market		
quality, and	Changes in net working capital	Momentum,	Distance to high
industry	Net operating assets	reversals,	Long-term reversals
	Industry concentration	and	Maximum daily return
		seasonality	Momentum
Profitability	Profit margin		Intermediate momentum
	Return on assets		Seasonality
	Return on equity		Seasonal reversals
	Gross profitability		Short-term reversals
	Earnings to price		Momentum and reversals
	QMJ profitability		
	Operating profitability	Other price	Amihud's illiquidity
	Cash-based profitability	and volume	Market beta
			Firm age
Distress	Z-score		Idiosyncratic volatility
and	O-score		Firm size
leverage	Financial distress		Nominal price
	Piotroski's F-score		High-volume return premium
	Operating leverage		Share volume
	Leverage		Coskewness

Finance is tricky because the signal is so weak !!

```
ddf = data.frame(x=TrxI,y = Rd[,2])
lmall = lm(y~.,ddf)
summary(lmall)

.....

Residual standard error: 0.112 on 1253690 degrees of freedom
Multiple R-squared:  0.003772,Adjusted R-squared:  0.003723
F-statistic: 76.57 on 62 and 1253690 DF,  p-value: < 2.2e-16

> sqrt(0.003772)
[1] 0.06141661
```

And this is in-sample, and things change over time.

Simple Effective Approach to Time-varying predictions:

R: cross section of returns, each month t, each firm in that month.
x: predictor variables used for R (measured at month $t-1$).

Our overall approach is the following:

- For each month t fit a model giving $\hat{R} = \hat{f}_t(x)$.
- Roll the fitted models: $\hat{f}_t^P(x) = \sum_{j=1}^{\nu} w_j \, \hat{f}_{t-j}(x)$.
- Check that $\hat{f}_t^P(x)$ has reasonable predictive performance.
- Inspect $\{\hat{f}_t^P\}$ to learn about the relationship, (e.g., what variables are used).
- Also consider $\hat{f}^A(x) = \frac{1}{N} \sum_{t=1}^{N} \hat{f}_t(x)$.

For example, we often use $\nu = 120$ (10 years), $w_j = 1/120$.

Choice of "Learner"

We have to fit a model each month so we want to use approaches that do not require a lot of tuning. In addition, our x variables are "messy" so we need methods that perform well in this case.

We focus on methods based on trees and ensembles of trees:

- Trees are capable of uncovering any kind of non-linearity and interaction.
- Trees handle messy x variables: they are invariant to monotonic transformations of the predictor variables.
- Single trees partition the x space into rectangular subsets somewhat reminiscent of what you obtain by sorting stocks into portfolios
- Ensembles of trees, in which many trees are combined to get an overall fit, are the best "off-the-shelf" models.
- We will use Random Forests and BART (Bayesian Additive Regression Trees) which is an ensemble method related to boosting. Generally, BART requires less tuning than other boosting type approaches. Random Forests is well known for performing well with minimal tuning.

We ran default BART and default random forests.

Default BART for each month:

```
##################################################
#fit a bart at each month (date)
library(BART)
nd=5000;burn=10000

set.seed(99)
rollBartTime=system.time({
rollBart = foreach(i=1:nmo) %dopar% {
   cat("&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&",i,"\n")
   ii = (dates == months[i])
   temp = wbart(TrxI[ii,],Rd[ii,2], ndpost=nd, nskip=burn,printevery=1000,
            nkeeptrain=0,nkeeptest=0,nkeeptestmean=0,
                   nkeeptreedraws=1000,rm.const=FALSE)
   list(i=i,mod=temp)
}
})
```

We ran default BART model specifications, but apparently we had to run it quite a while to find the signal.

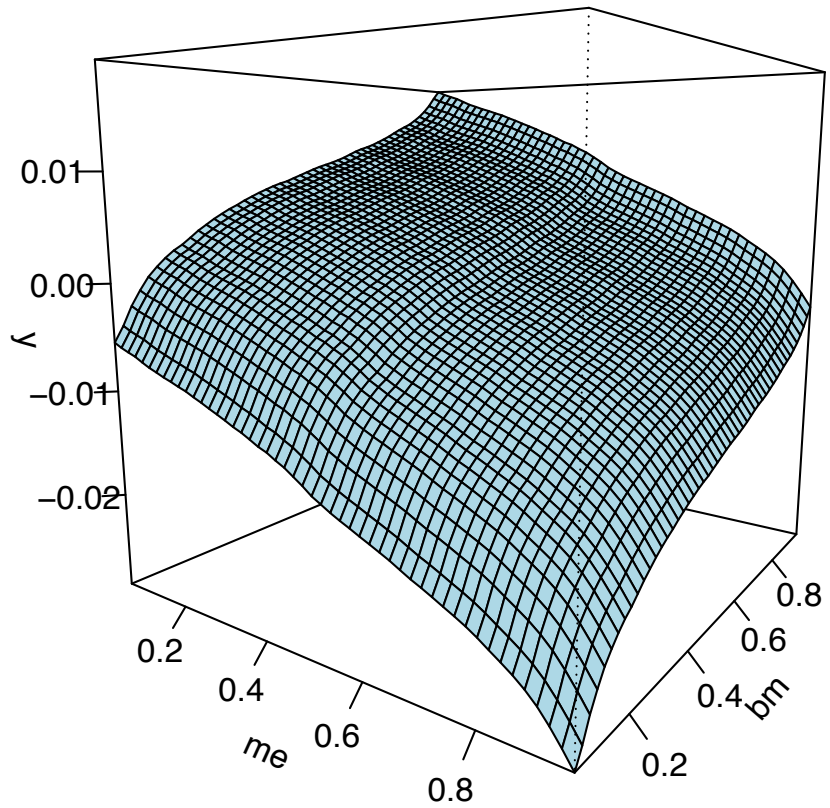
Each month just as a couple thousand very noisy observations !!!

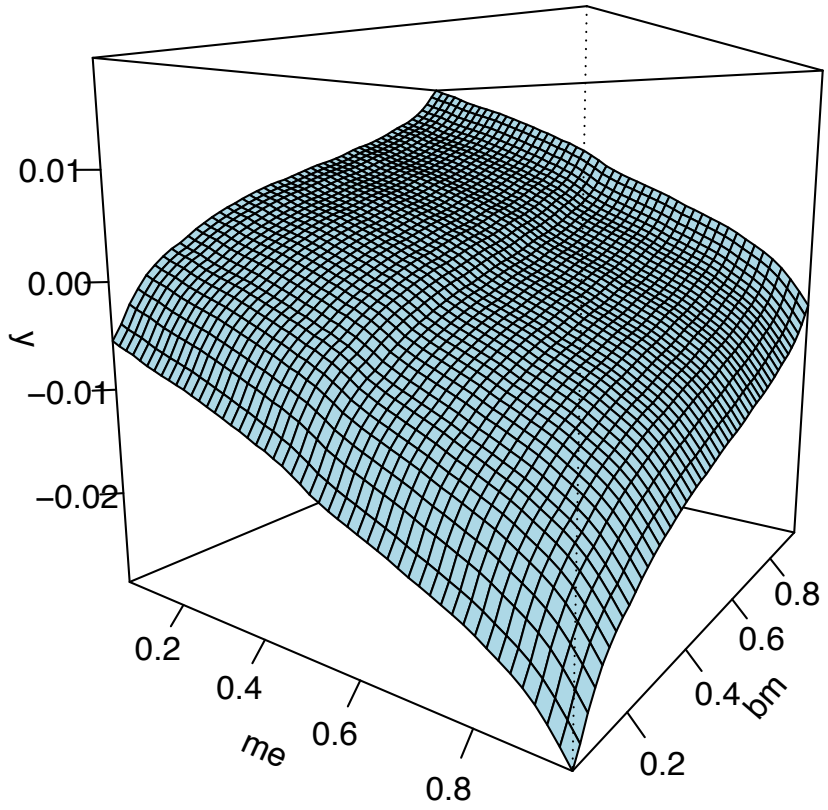
Our goal is to have some understanding of what the non-linear fitted relationship is.

With a two-dimensional x, we can plot.

E(R) vs
x1: me = market equity
x2: bm =book-to-market.

Hard in high dimensions!!!





- Looks pretty linear for most of the middle of the x=(me,bm) predictor space.
- big me and small bm really interact to give you low returns.
- A little non-linear upturn for big bm, especially at small me.
  At big me, small bm , there is a *dusty corner*.
- nonlinearity for large bm across a range of return values.

Note:

Most of the methods could be used with estimates of $E(R \mid x)$ from any learner.

For example, Gu, Kelly and Xiu have some interesting results with neural nets.

Most of our results just examine the fit $E(R \mid x)$, but we are working on capturing the uncertainty.

2. Predictability

Is there any predictive ability?

Are the Machine Learners any better than linear?

Stacked Correlations

Stack all the R for each month and all the out-of-sample $\hat{R}$ for each month and compute the simple pearson correlations.

rf is Random Forests.
bart is BART.

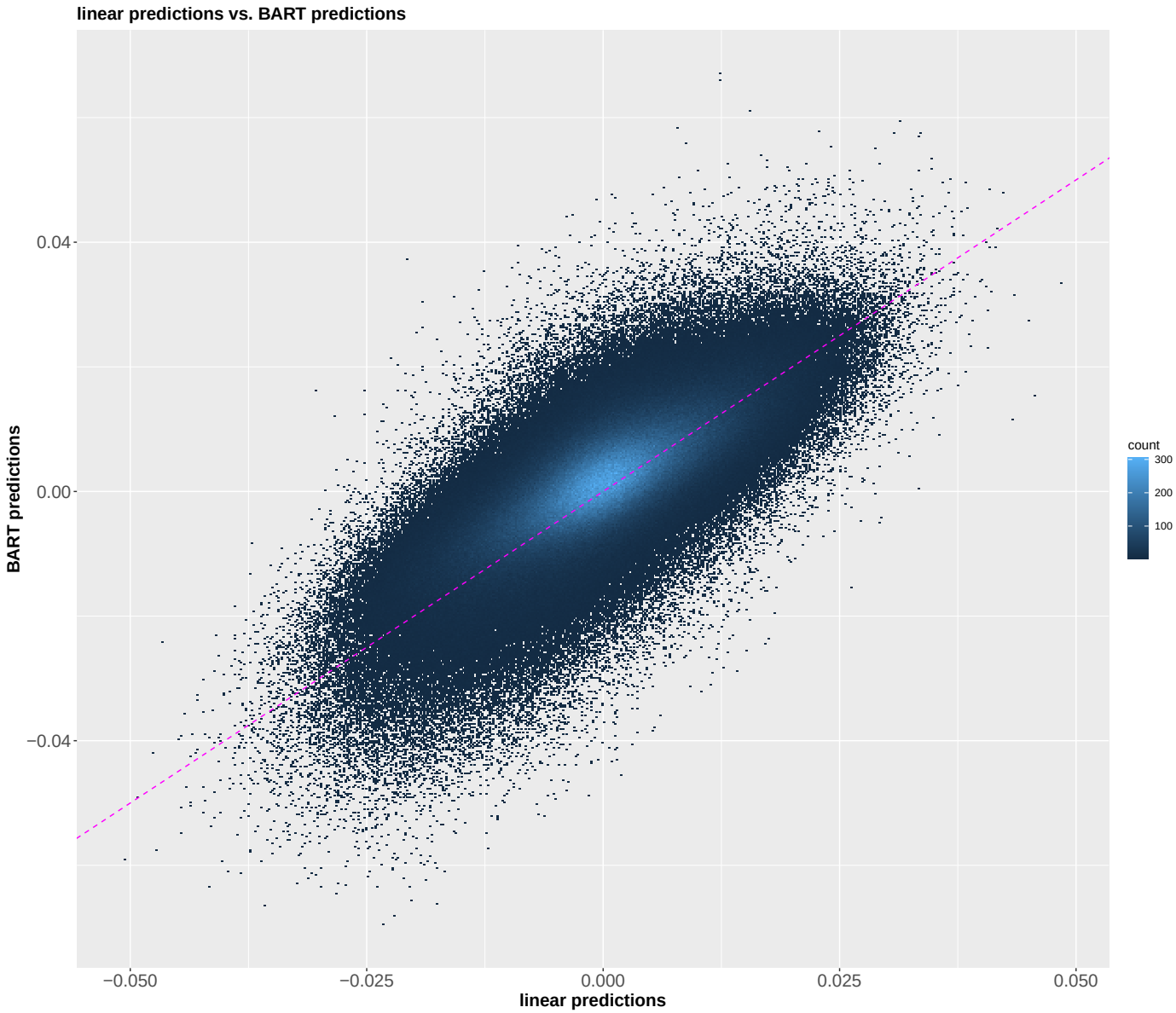
*15 uses just 15 variables we got from our variable selection.

*3 just uses "ME" "BE.ME" "r_12.2"

```
> round(cor(omat),3)
              oR olinear3  orf3 obart3 olinear   orf obart olinearv15 orfv15 obartv15
oR         1.000    0.027 0.024  0.027   0.048 0.040 0.054      0.046  0.051    0.056
olinear3   0.027    1.000 0.781  0.809   0.461 0.475 0.377      0.578  0.494    0.444
orf3       0.024    0.781 1.000  0.899   0.371 0.465 0.369      0.456  0.496    0.441
obart3     0.027    0.809 0.899  1.000   0.393 0.456 0.393      0.479  0.487    0.469
olinear    0.048    0.461 0.371  0.393   1.000 0.609 0.753      0.889  0.704    0.737
orf        0.040    0.475 0.465  0.456   0.609 1.000 0.698      0.631  0.805    0.672
obart      0.054    0.377 0.369  0.393   0.753 0.698 1.000      0.713  0.737    0.787
olinearv15 0.046    0.578 0.456  0.479   0.889 0.631 0.713      1.000  0.767    0.791
orfv15     0.051    0.494 0.496  0.487   0.704 0.805 0.737      0.767  1.000    0.851
obartv15   0.056    0.444 0.441  0.469   0.737 0.672 0.787      0.791  0.851    1.000
```

BART is more like linear and rf!!??

BART predictions compared to linear:



We looked at various ways to check there is some "BART predictability".

```
> lmf = lm(oR ~ olinear + obart + orf,omatDf)
summary(lmf)

Call:
lm(formula = oR ~ olinear + obart + orf, data = omatDf)

Residuals:
    Min      1Q  Median      3Q     Max
-1.0820 -0.0563 -0.0038  0.0509  6.3056

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0008179  0.0001372  -5.960 2.52e-09 ***
olinear      0.2212042  0.0196362  11.265  < 2e-16 ***
obart        0.4710455  0.0193945  24.288  < 2e-16 ***
orf          0.0462427  0.0231940   1.994   0.0462 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1153 on 1114194 degrees of freedom
Multiple R-squared:  0.002995,Adjusted R-squared:  0.002993
F-statistic:  1116 on 3 and 1114194 DF,  p-value: < 2.2e-16
```

Juhani:

```
The results show that adding the BART factor always helps.
This is not too surprising given the results in (1)
| those results show that the BART factor
contains information not found in the other factors.
```

3. Fit-the-Fit

We want to understand the function $\hat{R} = \hat{f}(x)$.

We create the data

$$(x, \hat{R} = 100\,\hat{f}(x))$$

where $\hat{f}$ is the $\hat{f}_t^P$ based on the BART models.

We then fit trees of various sizes to this "data".

It is easier for us to capture interpretable features from the tree than from the rolled BART predictions.

Since our predictions are actually time-varying, we will do it a second way as well.

We include time as variable when we fit-the-fit and fit trees to the data:

$$((x, t), \hat{R} = 100\,\hat{f}(x))$$

where t for an observation in month j is $j/684$.

So, this is a little different from a basic "fit-the-fit" because we are looking at time varying predictions rather than just a single $\hat{f}$ and we include t as a predictor in fitting a single tree.

1,114,198 "observations".

Without time in the "x" we are averaging over time and things are easier to understand.

With time in, we can get as sense of how the predictions are varying over time and get a more precise time specific feel for what is going on.

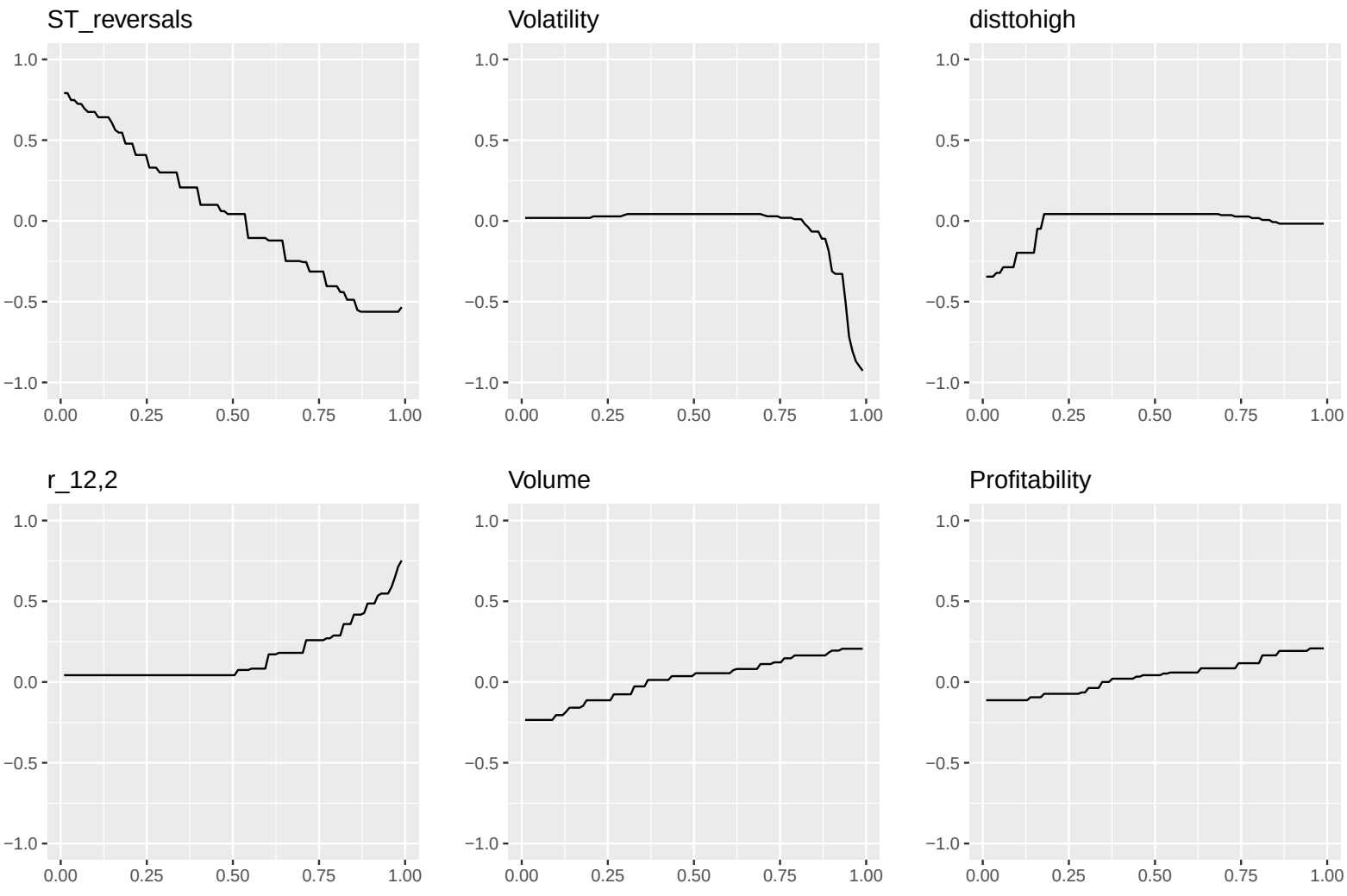
We actually need to do a lot more graphics to capture the time varying nature of our model.

For example, we should just divide our months up to chunks and look at the chunks separately but you will see we already have "enough" plots for today!!

A quick look ahead.

Plot predictions vs. some important predictor variables.

Not obvious what how this was done, or what it means, what about interactions !!



4. Basic fit-the-fit without T

Let's look at the results where we just fit trees to the fit-fit data without $T = \textit{time}$.

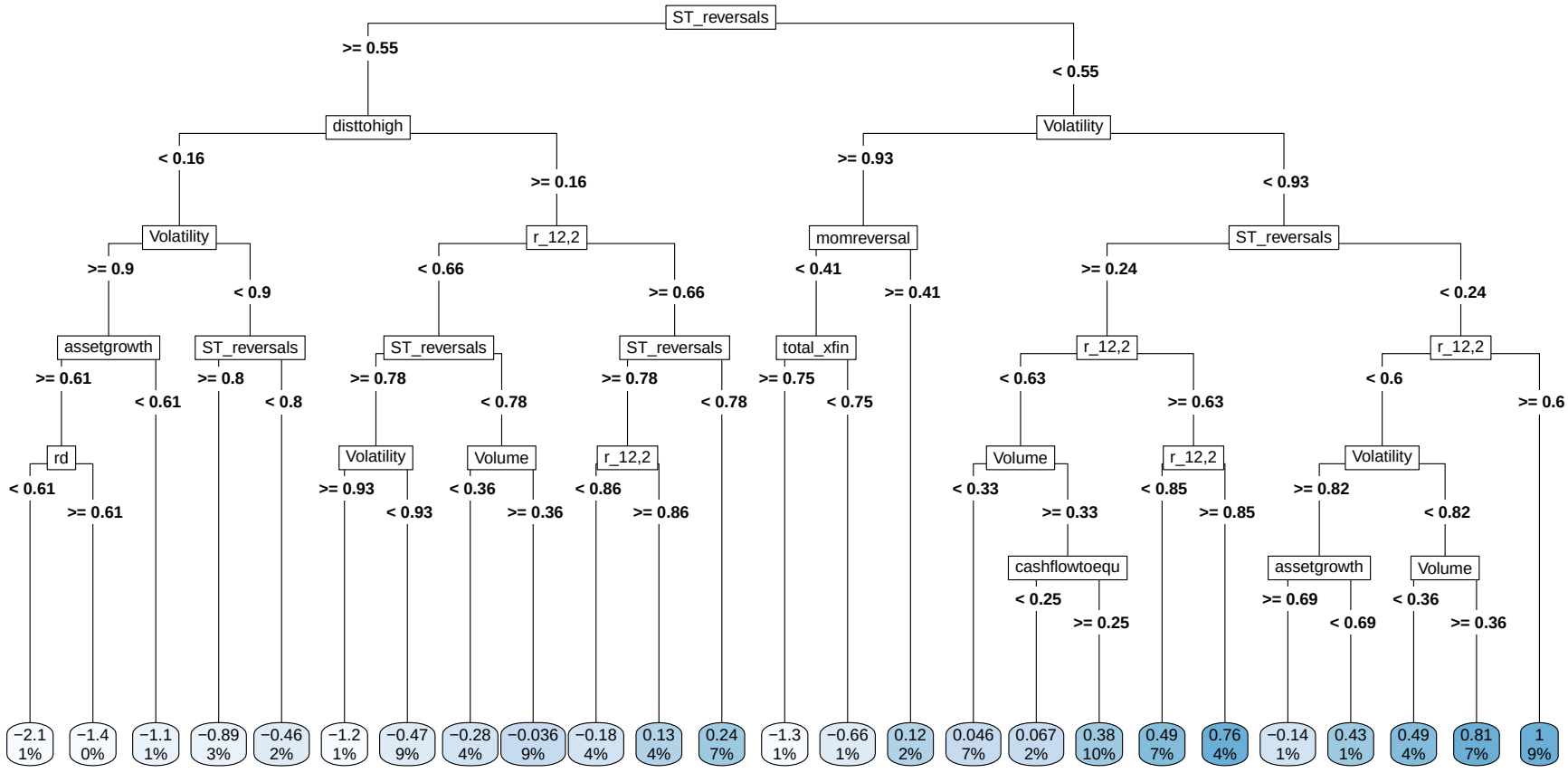
A basic issue immediately confronts us.

A small tree is easier to interpret but may be a crude representation of the fit (or time-varying predictions).

A big tree is harder to interpret but may be a more accurate representation.

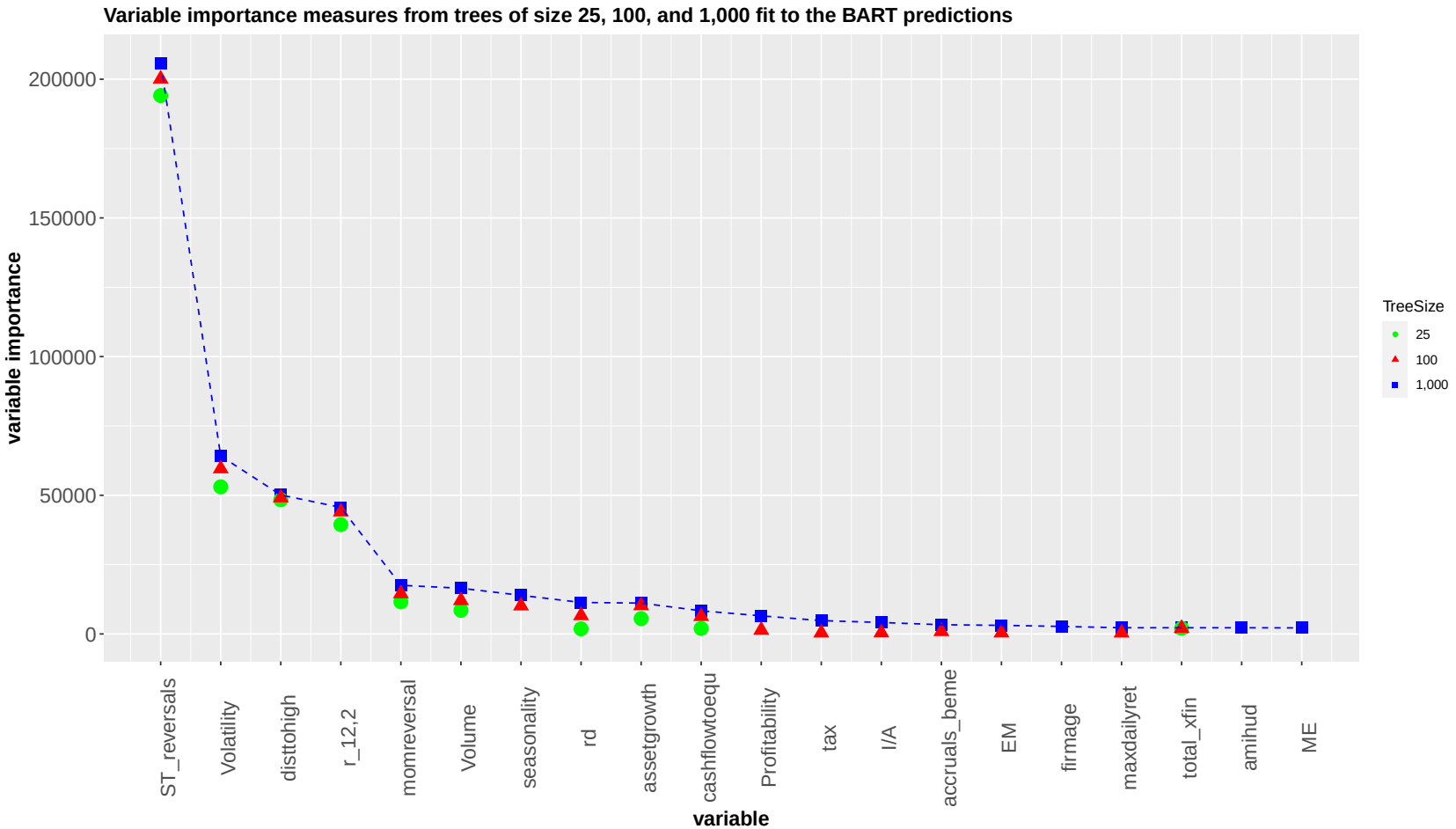
We fit trees with size= number of bottom nodes equal 25, 100, and 1,000.

- fit-fit tree with 25 bottom nodes
- software arranges the tree so that the mean predicted return goes up as you go left to right.



I love it.

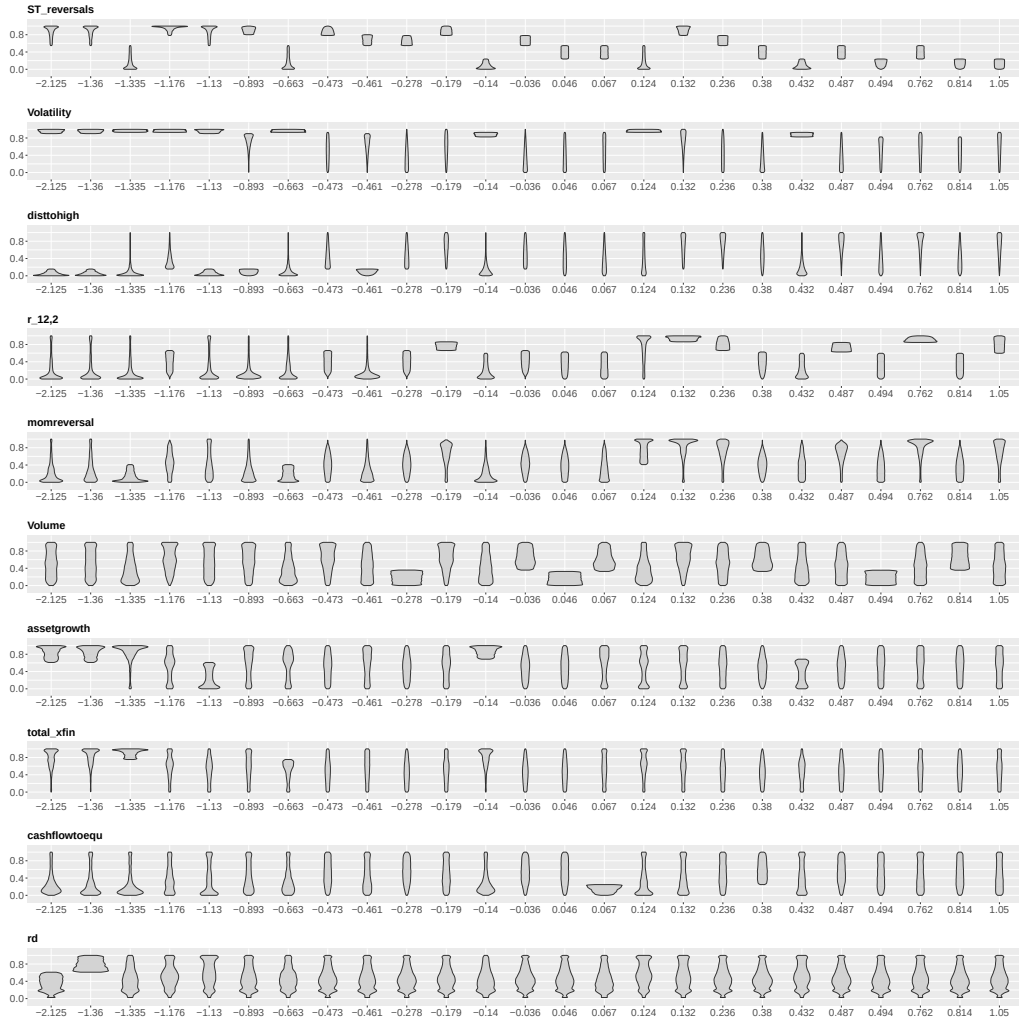
- classic Brieman variable importance from all three tree sizes.
- given by R package rpart.



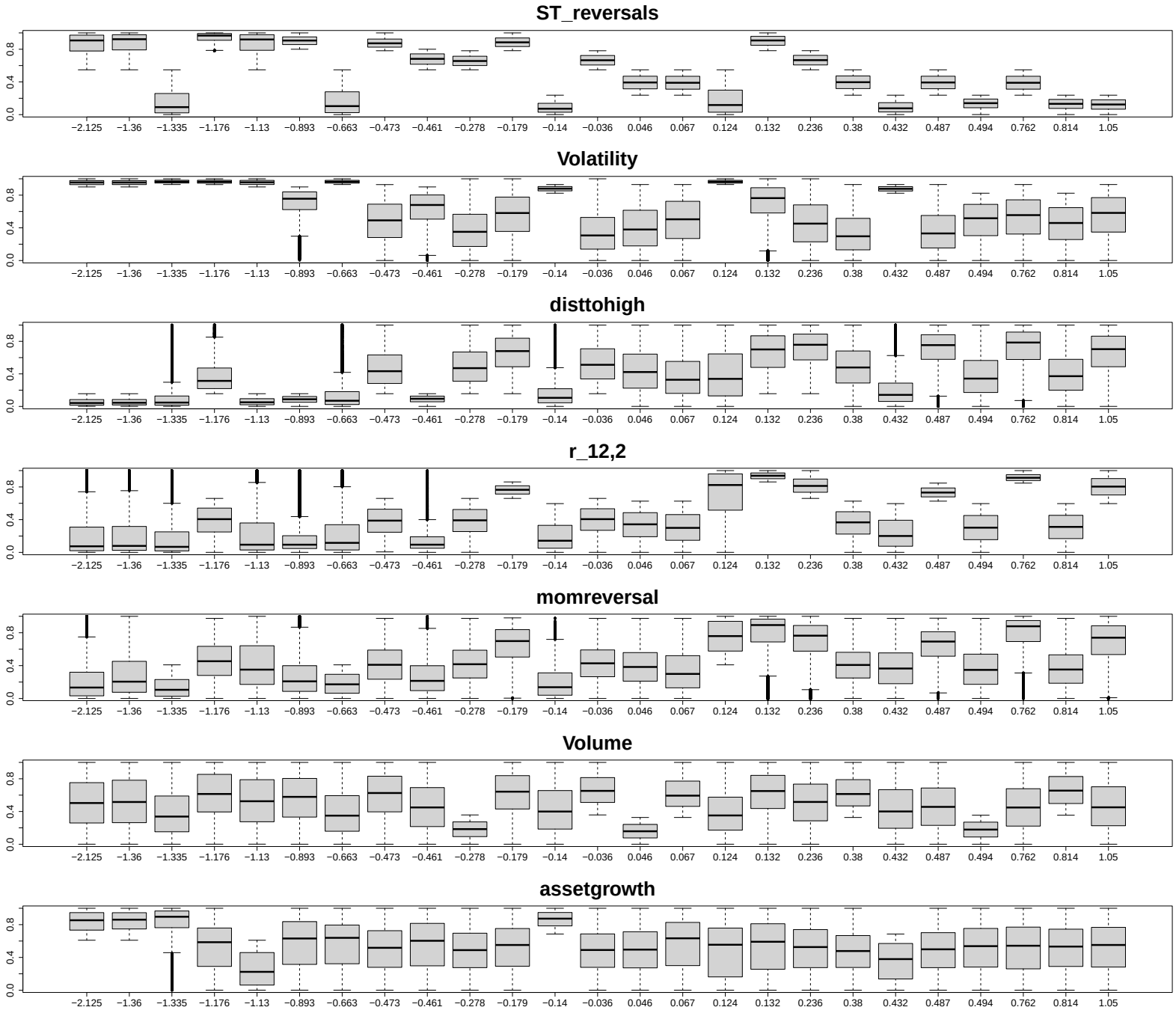
I love the tree but I also concede that even the 25 tree is hard to really take in.

To understand the tree we plot the distribution of the predictor variables in the bottom nodes.

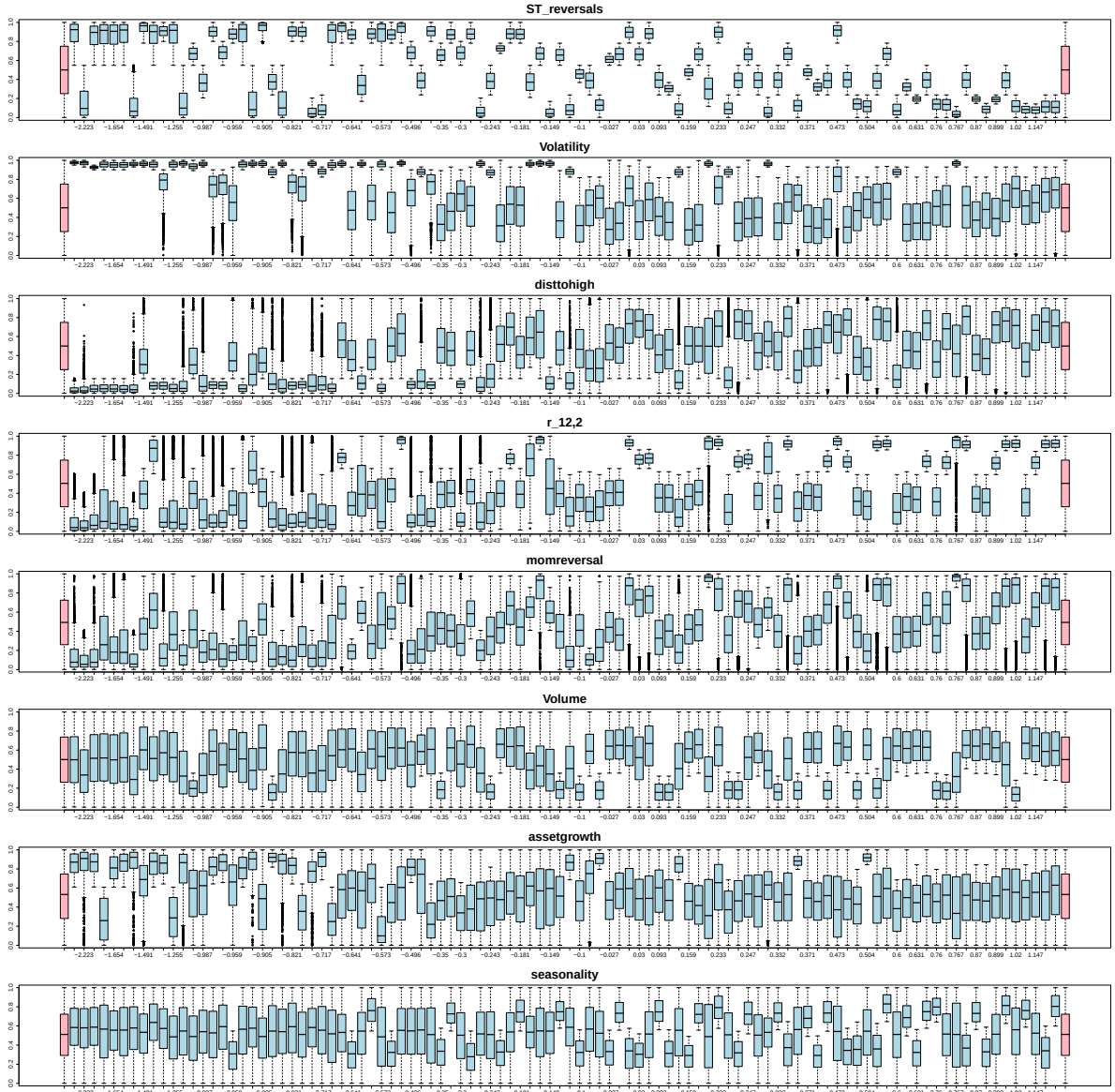
- tree with 25 bottom nodes.
- each row is a variable, rows sorted by variable importance from 25 tree.
- each column is a bottom node, sorted by mean prediction in a bottom node.
- each violin depicts the distribution of the row predictor in the column bottom node.



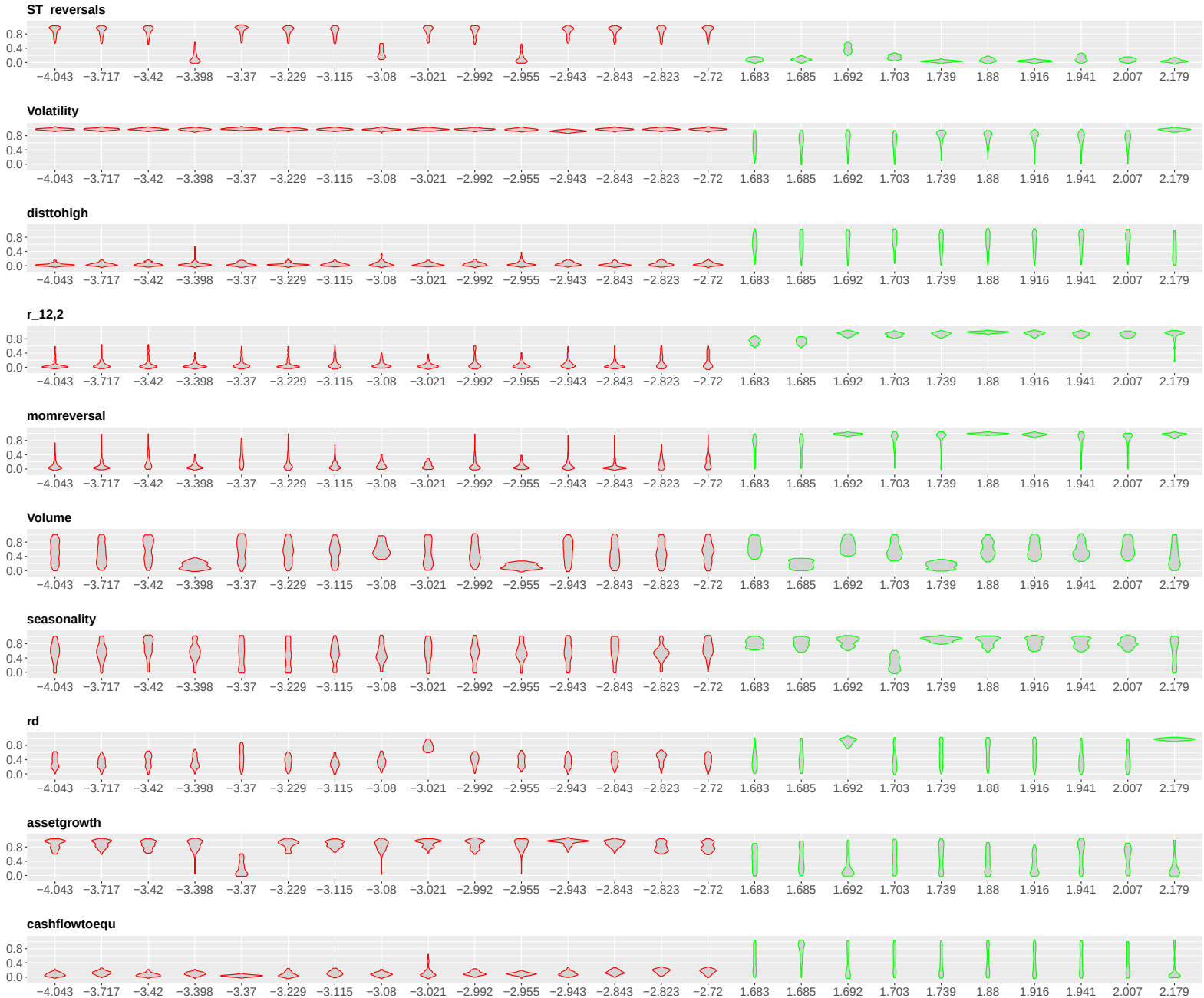
With boxplots instead of violins.



With boxplots, 100 size tree, marginal of predictor in red at each end of each row.



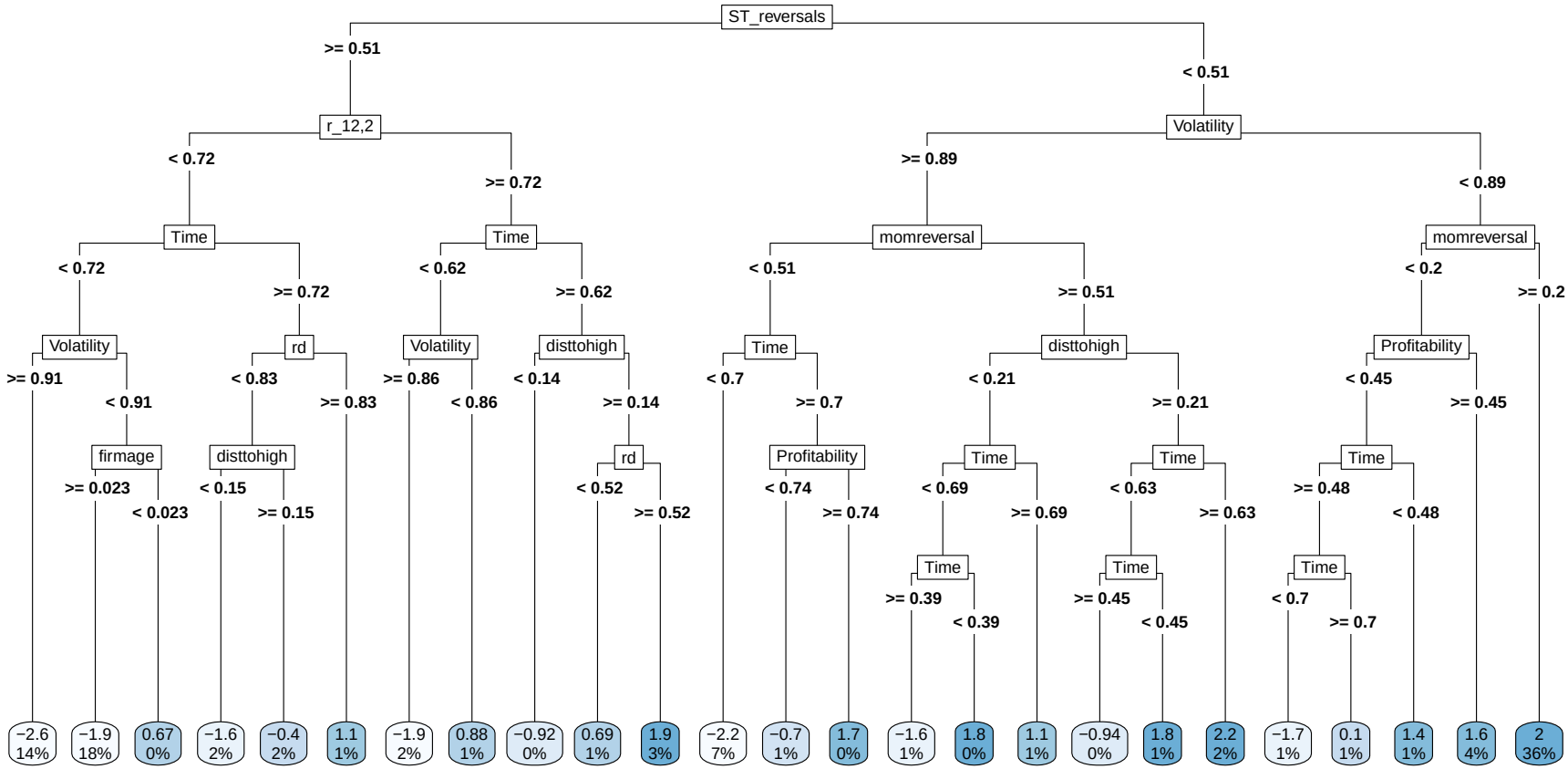
Violins, 1,000 bottom node tree. First 15 nodes with lowest mean, last 10 highest mean.



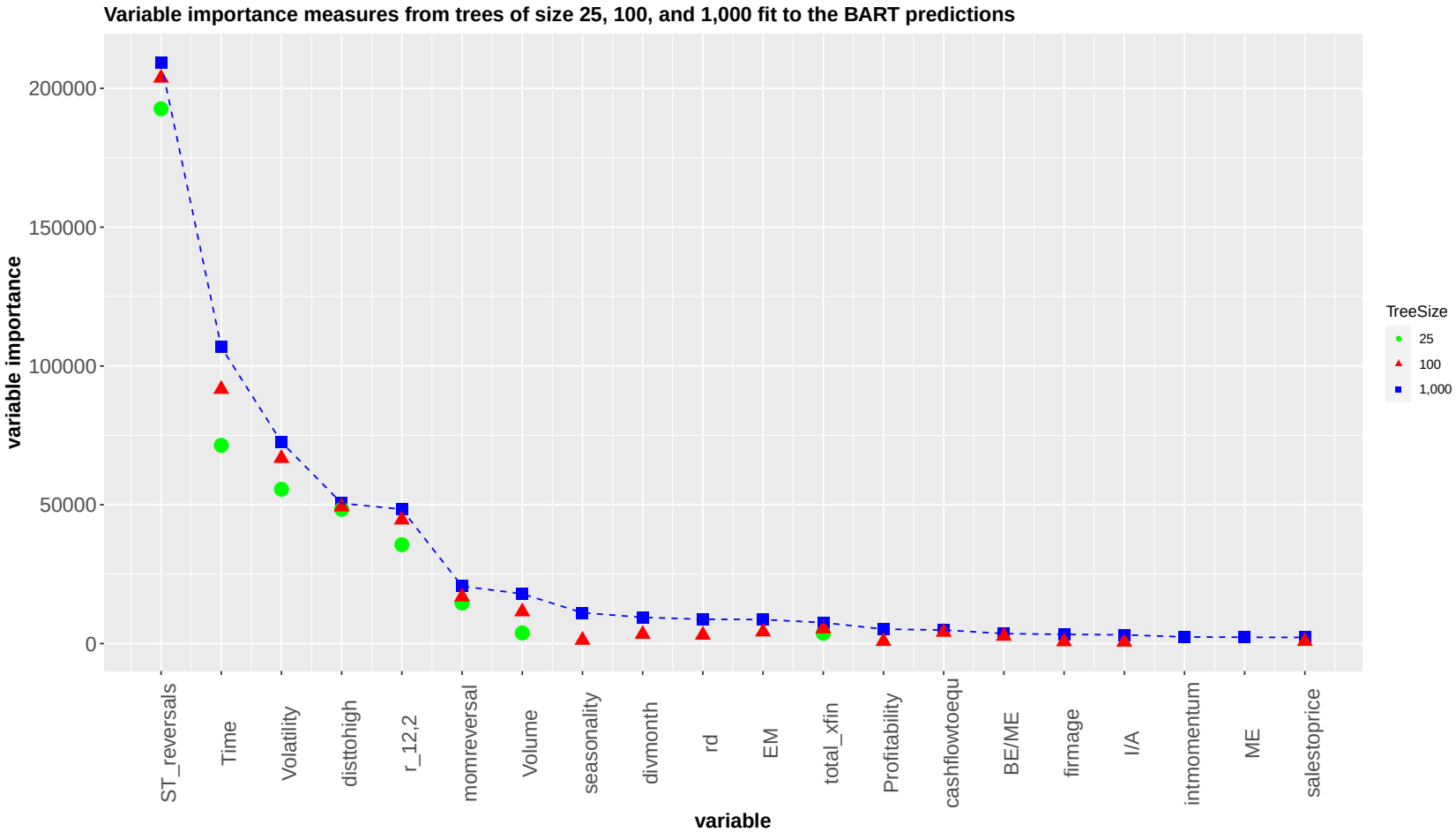
5. fit-fit with Time

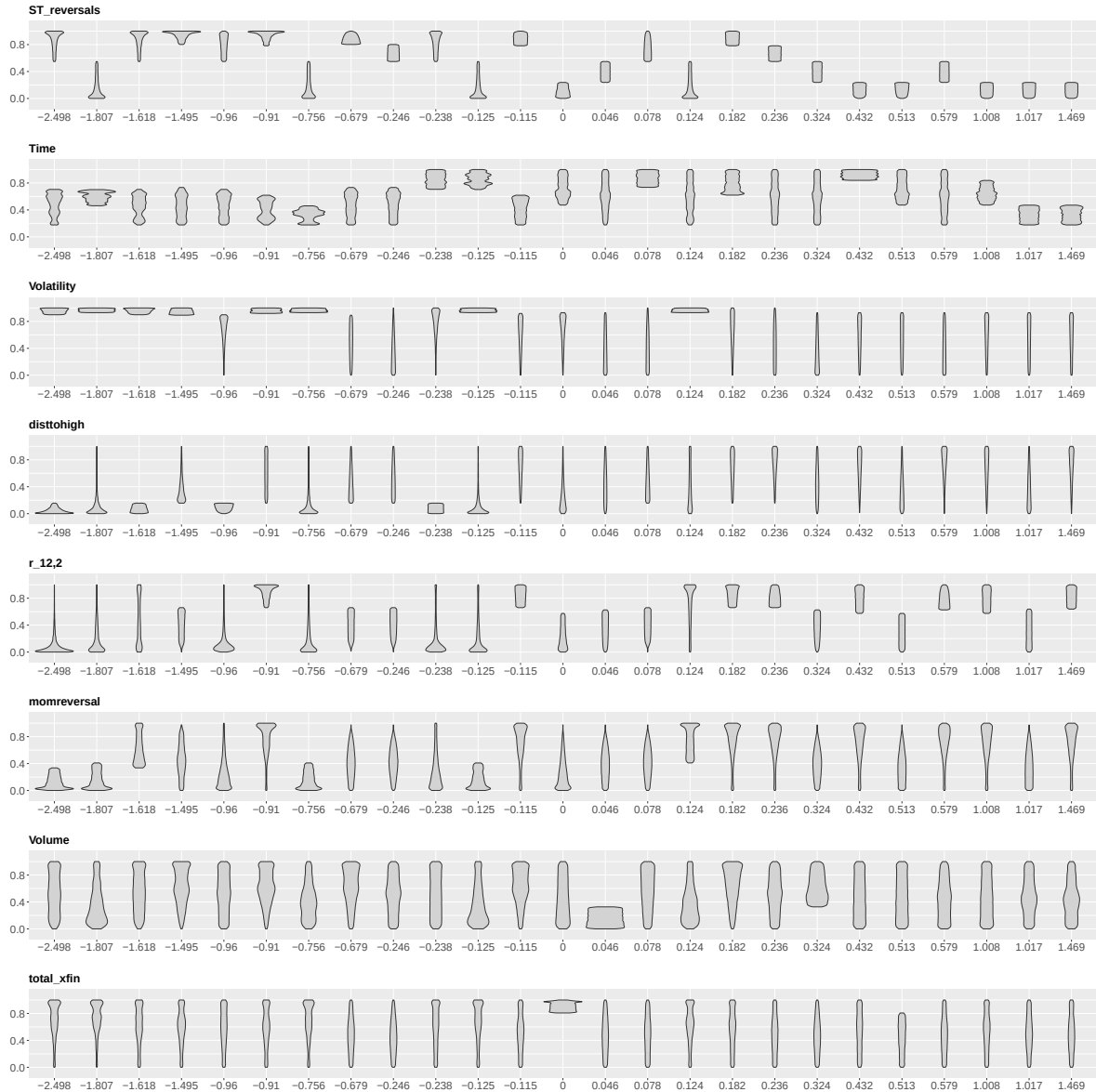
Same thing again, but this time we include time (month scaled to (0,1)) in the fit of the trees.

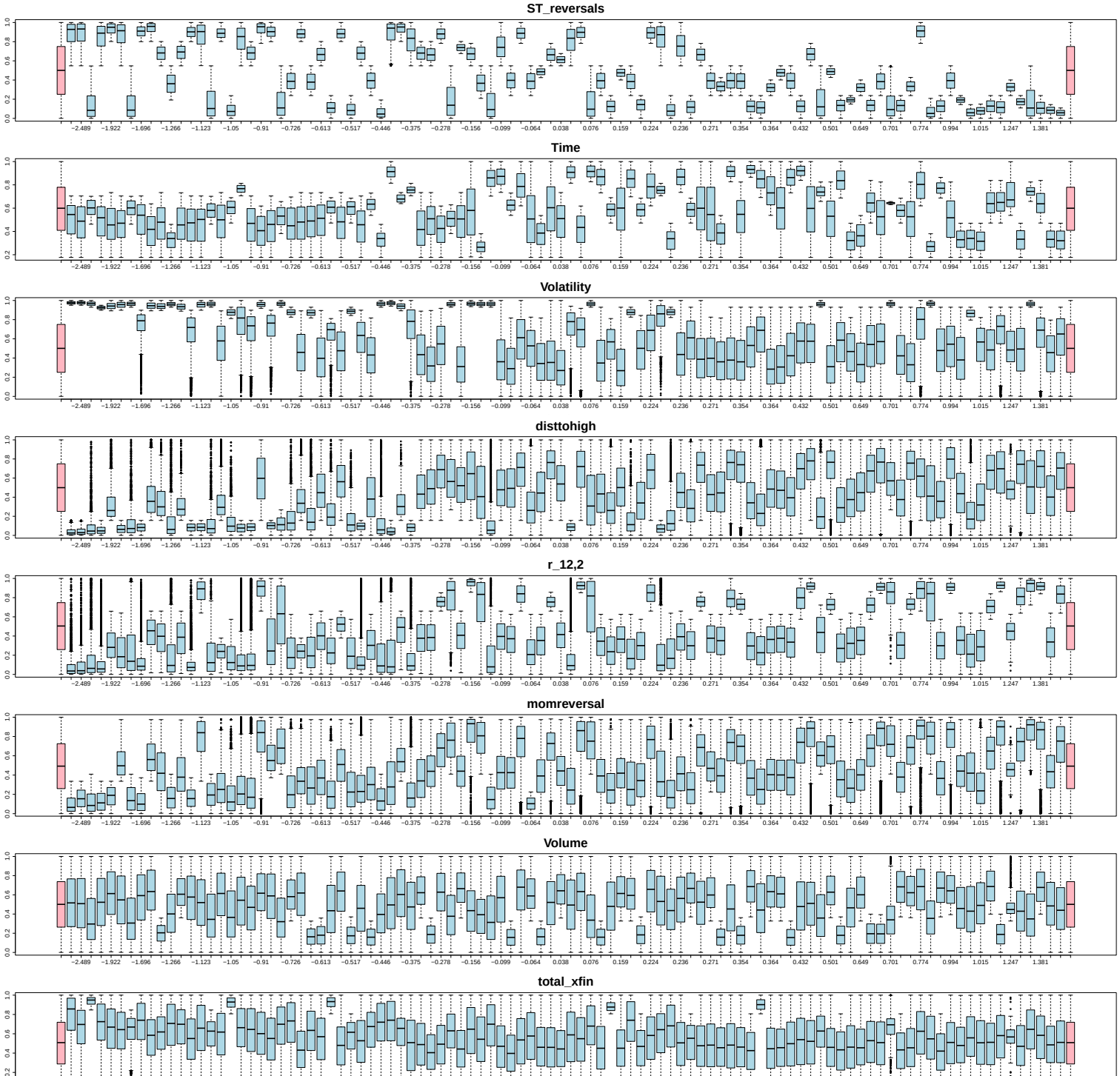
Fit-the-fit (actually fit the time varying predictions) Tree with 25 bottom nodes.

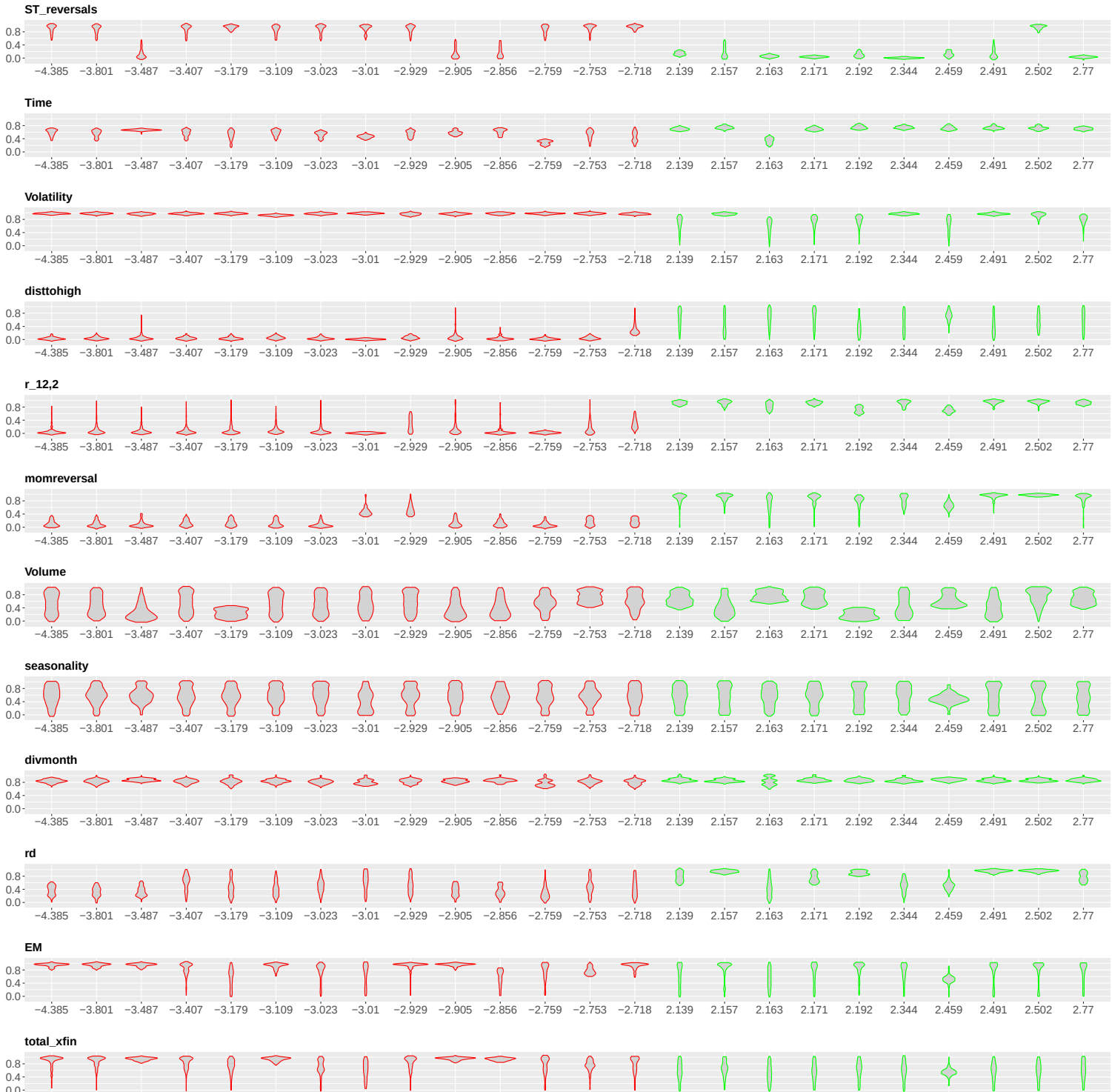


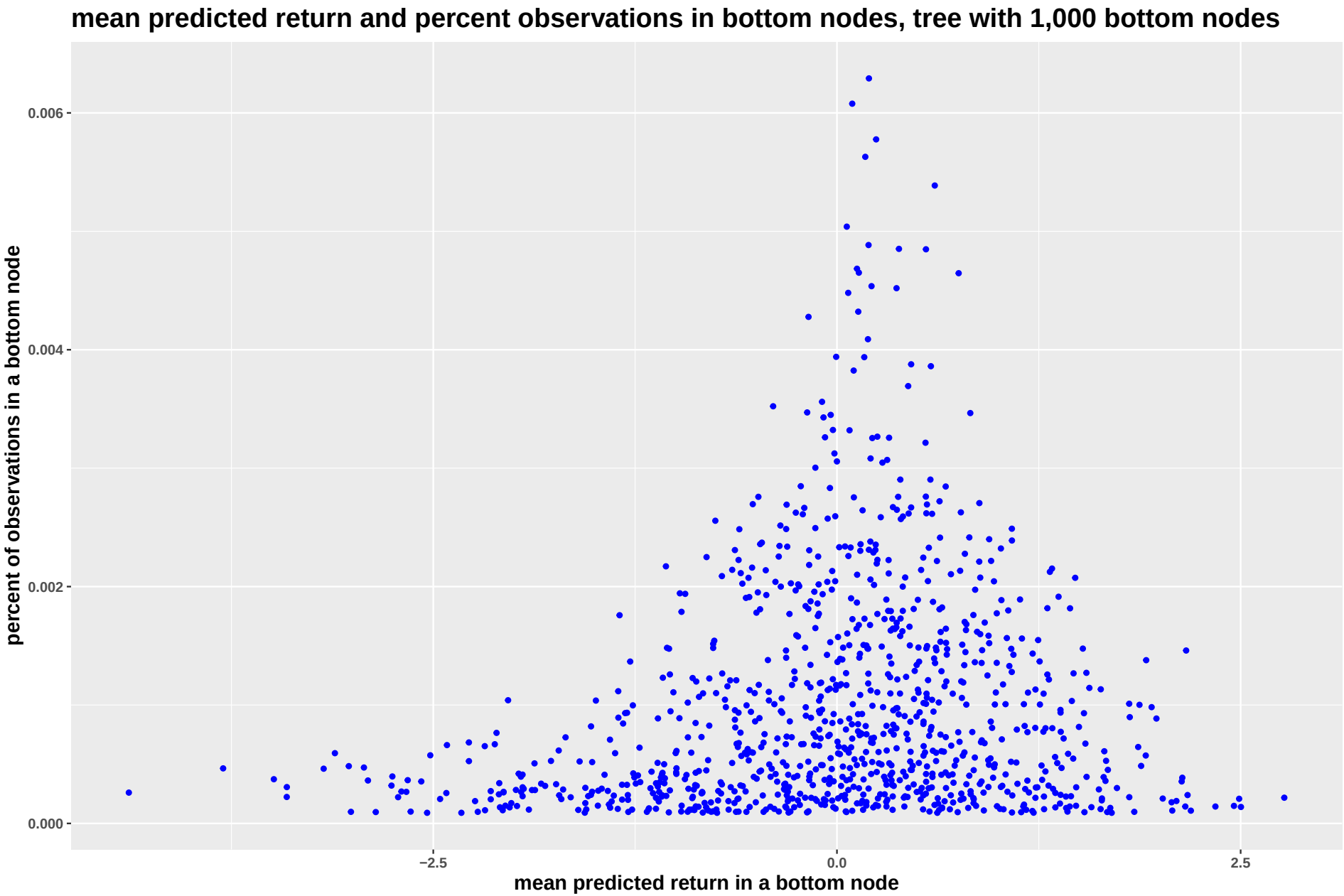

```
tree of size:  25
        R
 -2.49843 when ST_reversals >= 0.55 & Time <  0.71 & Volatility >= 0.90 &
      disttohigh <  0.16 & momreversal <  0.34
```


Variable importance measures from trees of size 25, 100, and 1,000 fit to the BART predictions
variable importance
200000
150000
100000
50000
0
ST_reversals
Time
Volatility
disttohigh
r_12,2
momreversal
Volume
seasonality
divmonth
rd
EM
total_xfin
Profitability
cashflowtoequ
BE/ME
firmage
I/A
intmomentum
ME
salestoprice
variable
TreeSize
25
100
1,000


ST_reversals
Time
Volatility
disttohigh
r_12,2
momreversal
Volume
total_xfin
0.8
0.4
0.0
−2.498 −1.807 −1.618 −1.495 −0.96 −0.91 −0.756 −0.679 −0.246 −0.238 −0.125 −0.115 0 0.046 0.078 0.124 0.182 0.236 0.324 0.432 0.513 0.579 1.008 1.017 1.469


ST_reversals
Time
Volatility
disttohigh
r_12,2
momreversal
Volume
total_xfin
−2.489 −1.922 −1.696 −1.266 −1.123 −1.05 −0.91 −0.726 −0.613 −0.517 −0.446 −0.375 −0.278 −0.156 −0.099 −0.064 0.038 0.076 0.159 0.224 0.236 0.271 0.354 0.364 0.432 0.501 0.649 0.701 0.774 0.994 1.015 1.247 1.381


ST_reversals
Time
Volatility
disttohigh
r_12,2
momreversal
Volume
seasonality
divmonth
rd
EM
total_xfin
0.8
0.4
0.0
−4.385
−3.801
−3.487
−3.407
−3.179
−3.109
−3.023
−3.01
−2.929
−2.905
−2.856
−2.759
−2.753
−2.718
2.139
2.157
2.163
2.171
2.192
2.344
2.459
2.491
2.502
2.77


mean predicted return and percent observations in bottom nodes, tree with 1,000 bottom nodes
percent of observations in a bottom node
0.006
0.004
0.002
0.000
−2.5
0.0
2.5
mean predicted return in a bottom node

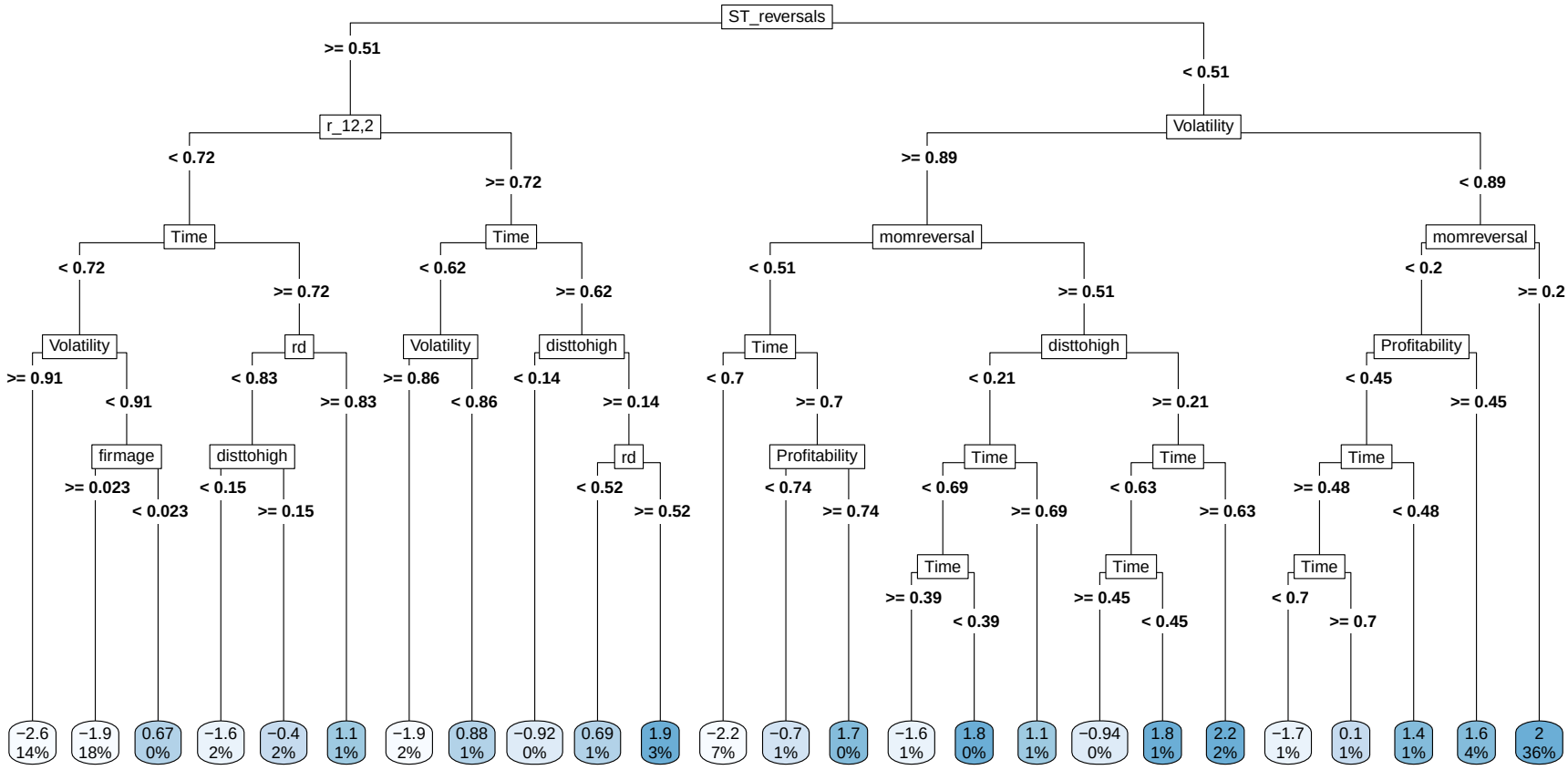
Searching for Dusty Corners

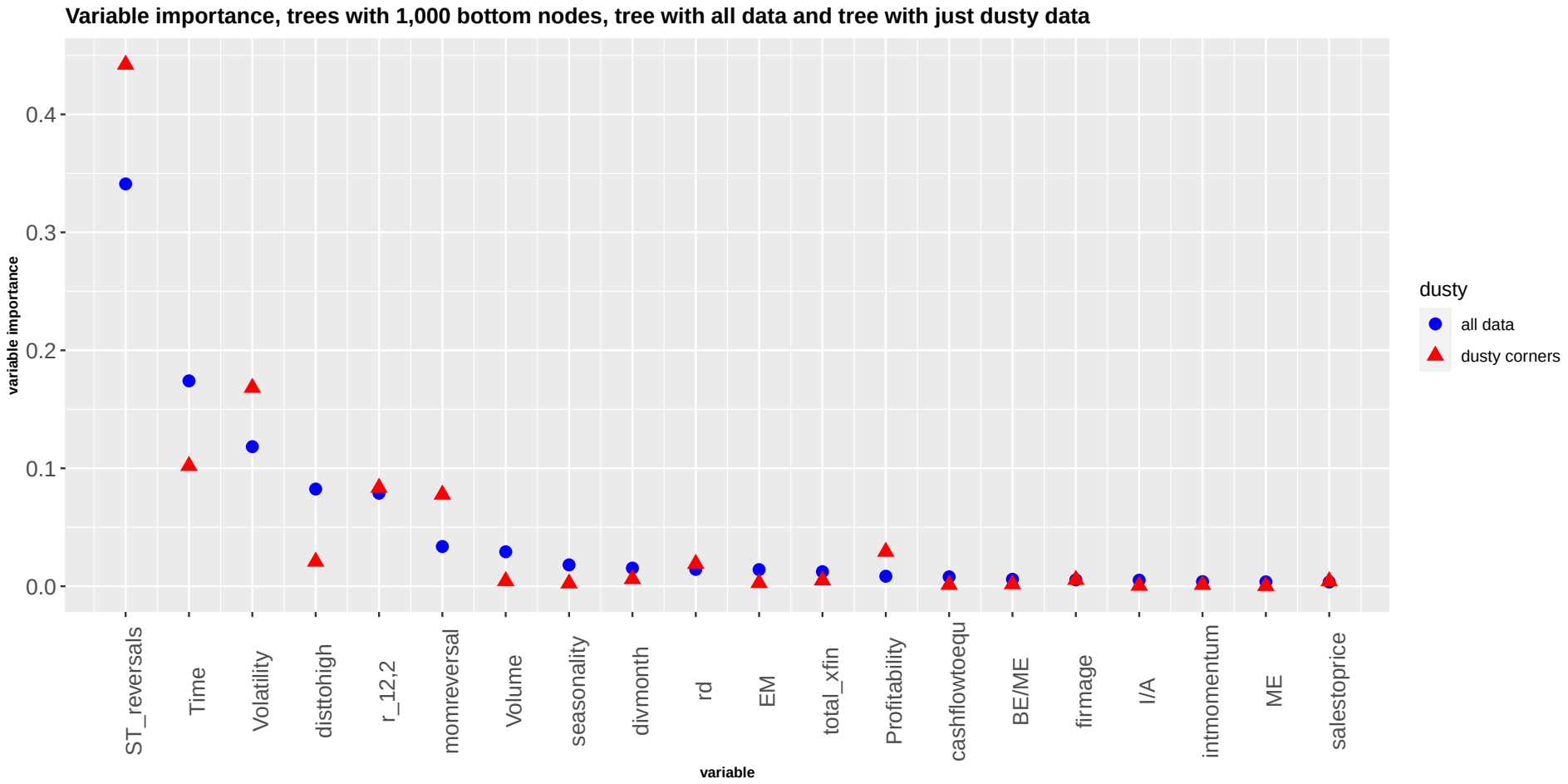
Things like the variable importance average over all all the observations.

To zero in in unusual returns (and dusty corners) we fit the trees using only observations with unusually large or small predicted returns.

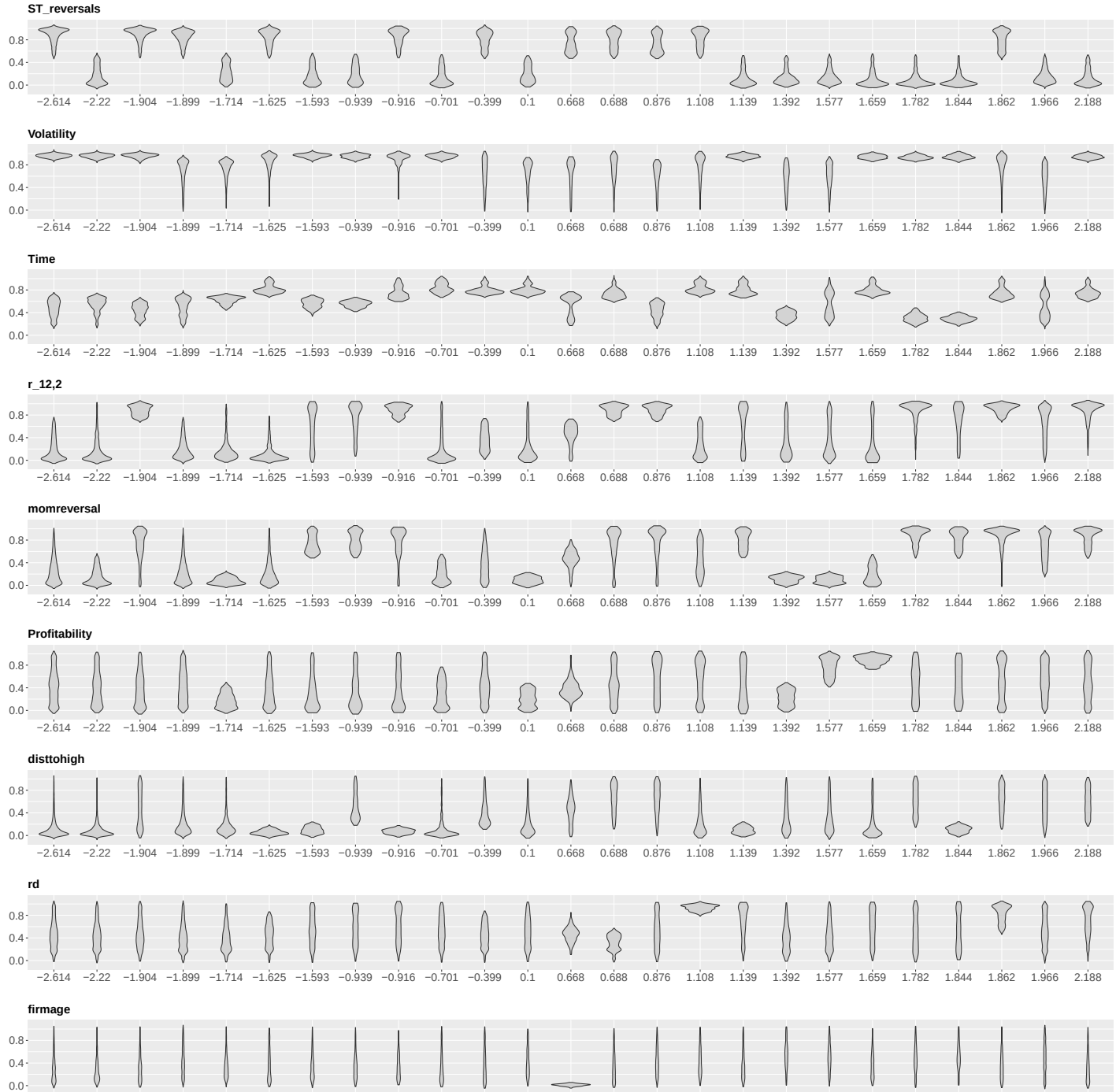
We call this the "dusty data".

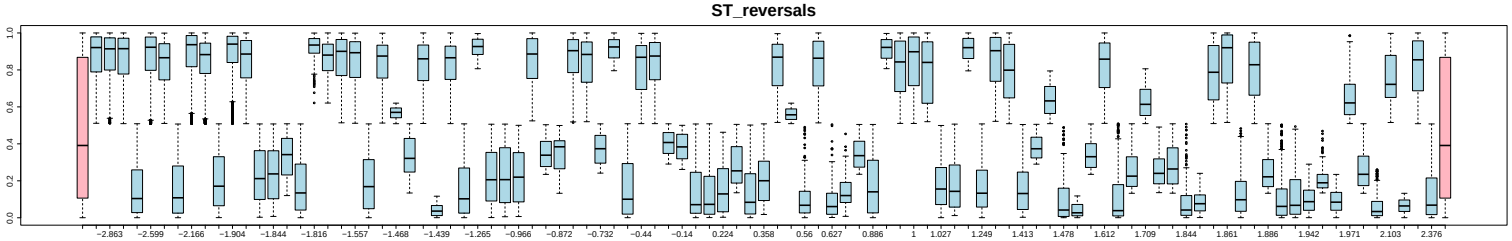
25 bottom nodes, just using dusty data.

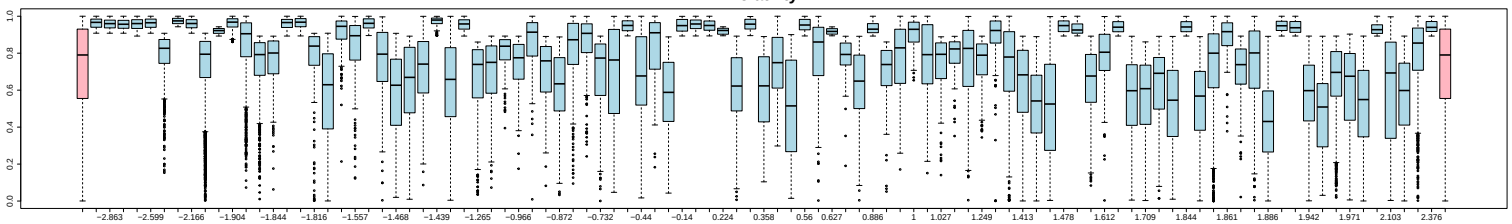


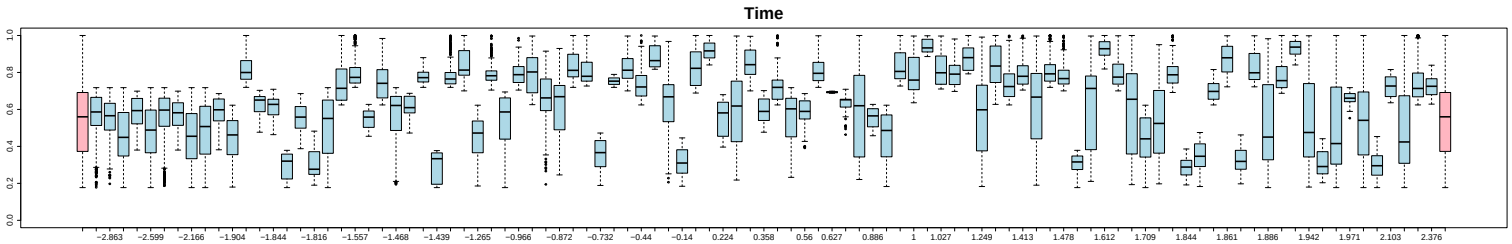

Variable importance, trees with 1,000 bottom nodes, tree with all data and tree with just dusty data
variable importance
0.0
0.1
0.2
0.3
0.4
ST_reversals
Time
Volatility
disttohigh
r_12,2
momreversal
Volume
seasonality
divmonth
rd
EM
total_xfin
Profitability
cashflowtoequ
BE/ME
firmage
I/A
intmomentum
ME
salestoprice
variable
dusty
all data
dusty corners

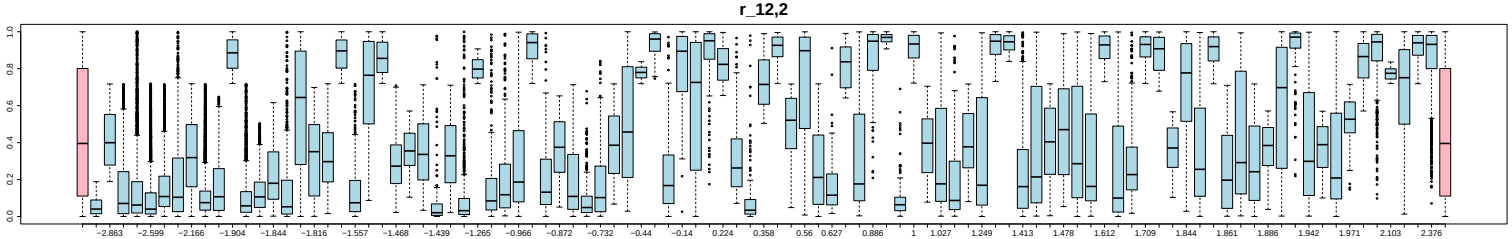
Dusty data, tree 25.

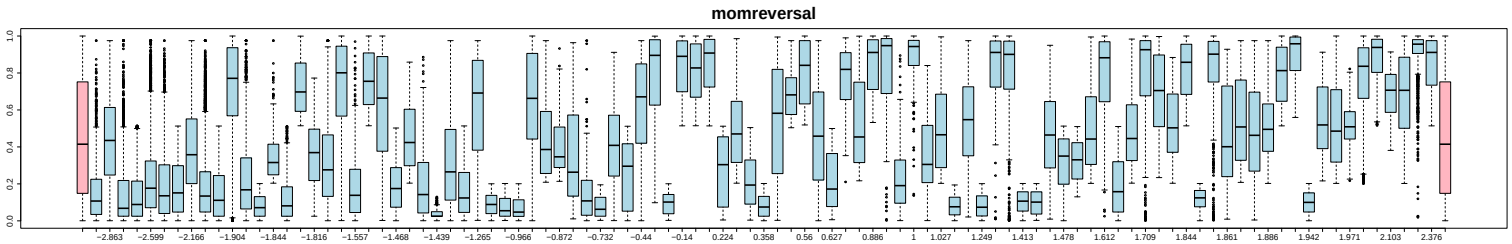


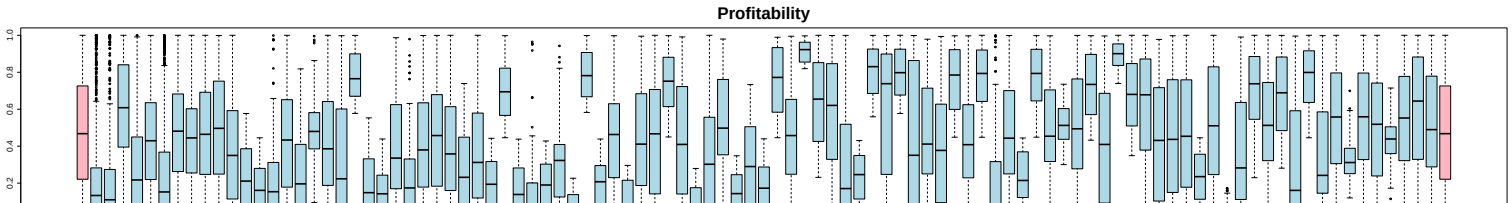

ST_reversals
0.0
0.2
0.4
0.6
0.8
1.0
−2.863 −2.599 −2.166 −1.904 −1.844 −1.816 −1.557 −1.468 −1.439 −1.265 −0.966 −0.872 −0.732 −0.44 −0.14 0.224 0.358 0.56 0.627 0.886 1 1.027 1.249 1.413 1.478 1.612 1.709 1.844 1.861 1.886 1.942 1.971 2.103 2.376


Volatility
0.0
0.2
0.4
0.6
0.8
1.0
−2.863 −2.599 −2.166 −1.904 −1.844 −1.816 −1.557 −1.468 −1.439 −1.265 −0.966 −0.872 −0.732 −0.44 −0.14 0.224 0.358 0.56 0.627 0.886 1 1.027 1.249 1.413 1.478 1.612 1.709 1.844 1.861 1.886 1.942 1.971 2.103 2.376


Time
0.0
0.2
0.4
0.6
0.8
1.0
−2.863 −2.599 −2.166 −1.904 −1.844 −1.816 −1.557 −1.468 −1.439 −1.265 −0.966 −0.872 −0.732 −0.44 −0.14 0.224 0.358 0.56 0.627 0.886 1 1.027 1.249 1.413 1.478 1.612 1.709 1.844 1.861 1.886 1.942 1.971 2.103 2.376


r_12,2
0.0
0.2
0.4
0.6
0.8
1.0
−2.863 −2.599 −2.166 −1.904 −1.844 −1.816 −1.557 −1.468 −1.439 −1.265 −0.966 −0.872 −0.732 −0.44 −0.14 0.224 0.358 0.56 0.627 0.886 1 1.027 1.249 1.413 1.478 1.612 1.709 1.844 1.861 1.886 1.942 1.971 2.103 2.376


momreversal
0.0
0.2
0.4
0.6
0.8
1.0
−2.863 −2.599 −2.166 −1.904 −1.844 −1.816 −1.557 −1.468 −1.439 −1.265 −0.966 −0.872 −0.732 −0.44 −0.14 0.224 0.358 0.56 0.627 0.886 1 1.027 1.249 1.413 1.478 1.612 1.709 1.844 1.861 1.886 1.942 1.971 2.103 2.376


Profitability
0.0
0.2
0.4
0.6
0.8
1.0

6. Seeing the Nonlinearity, T in the tree, Rolling fits

We saw before that the BART-based predictions are pretty highly correlated with the linear-based predictions.

Our previous fit-the-fit look at the BART-based predictions included the part that "looks linear".

We will regress the BART-based predictions on x *each month* and get the residuals.

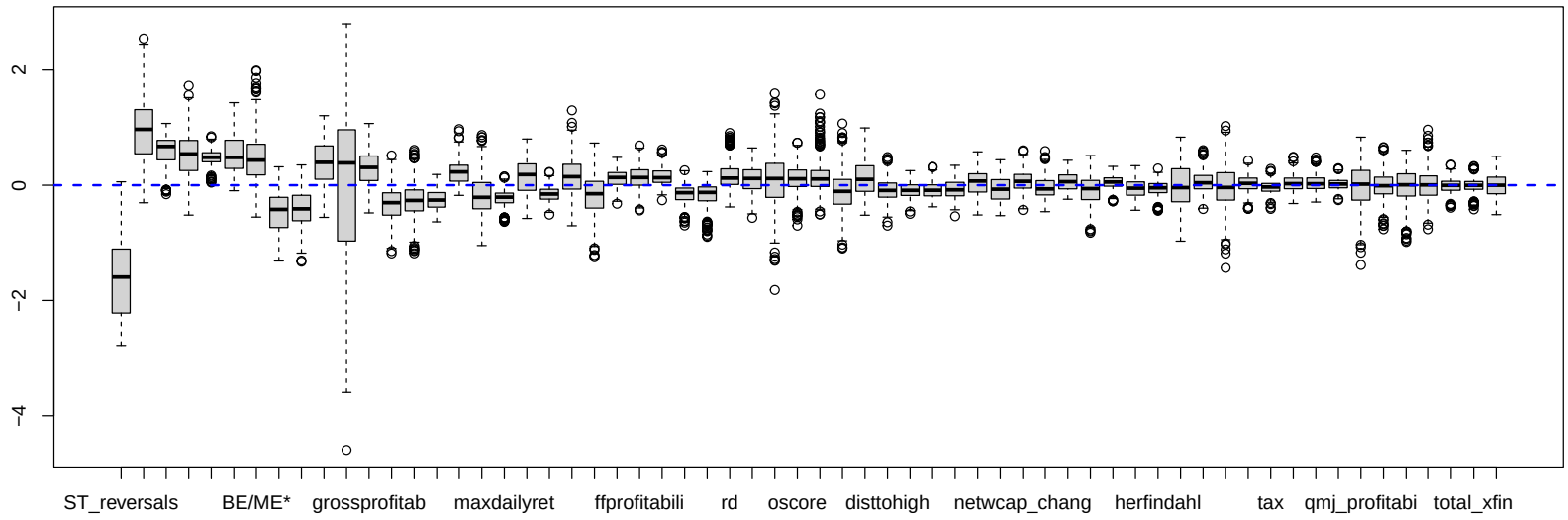
We think of the stacked residuals as the nonlinear part of our model.

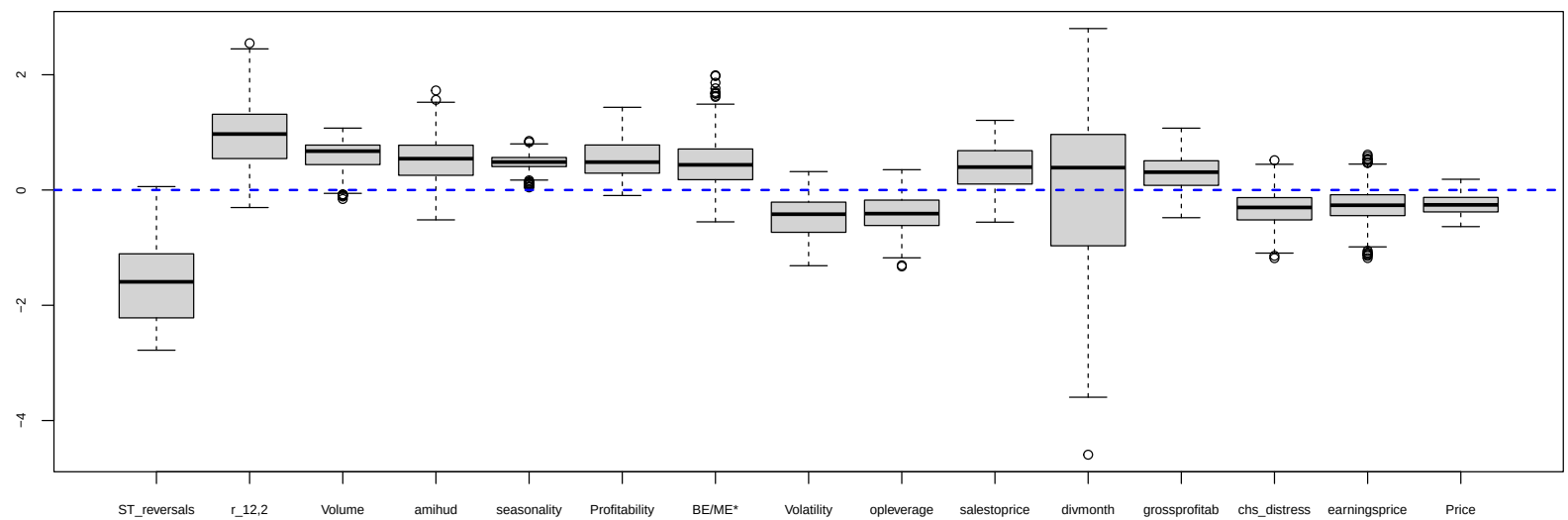
We use our single tree based fit-the-fit with the time variable included with the 62 x variables and the nonlinear part as the response.

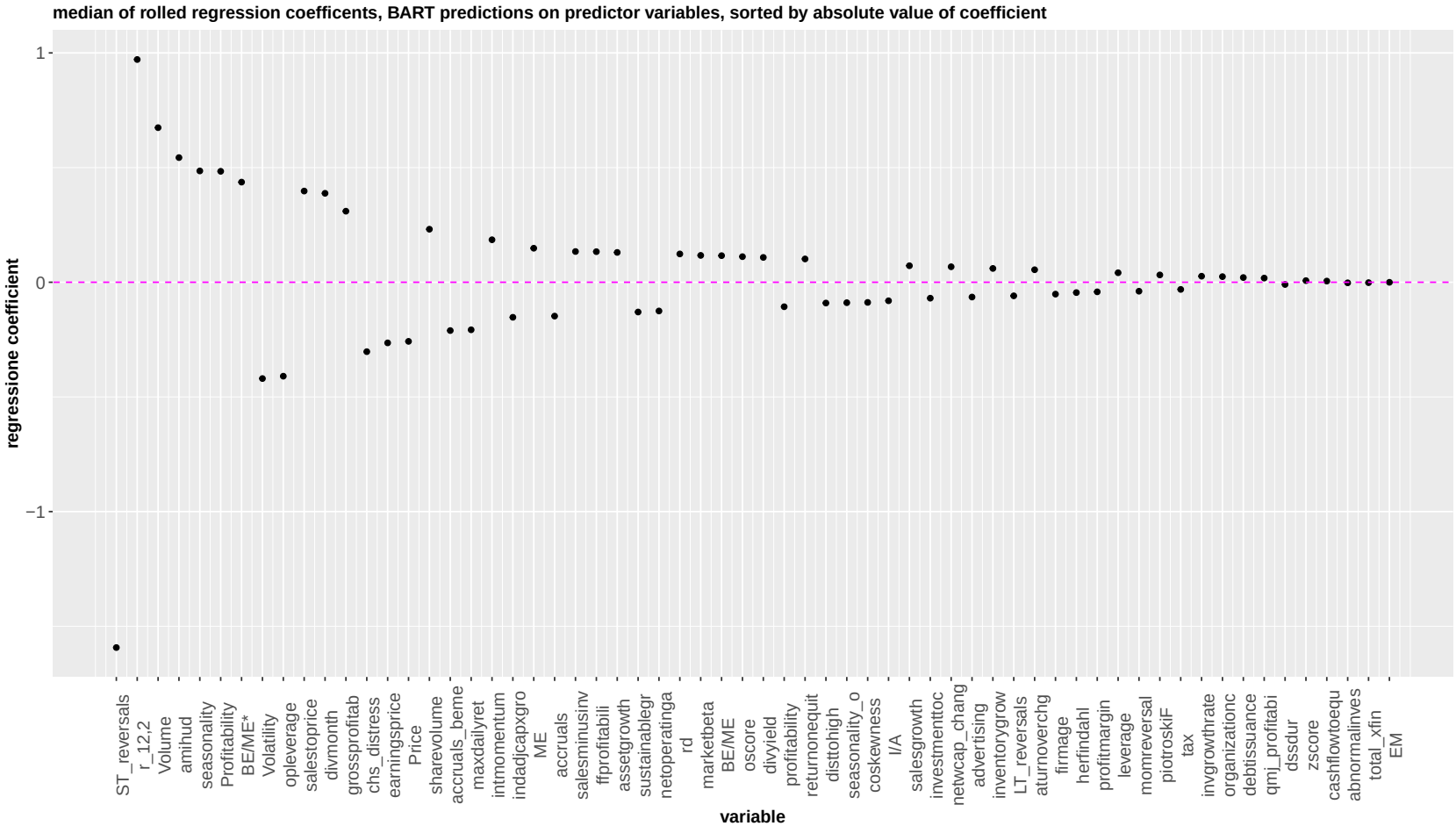
In the simpler "let's not worry about time" version we would just regress all the bart predictions on the x's without T=time.

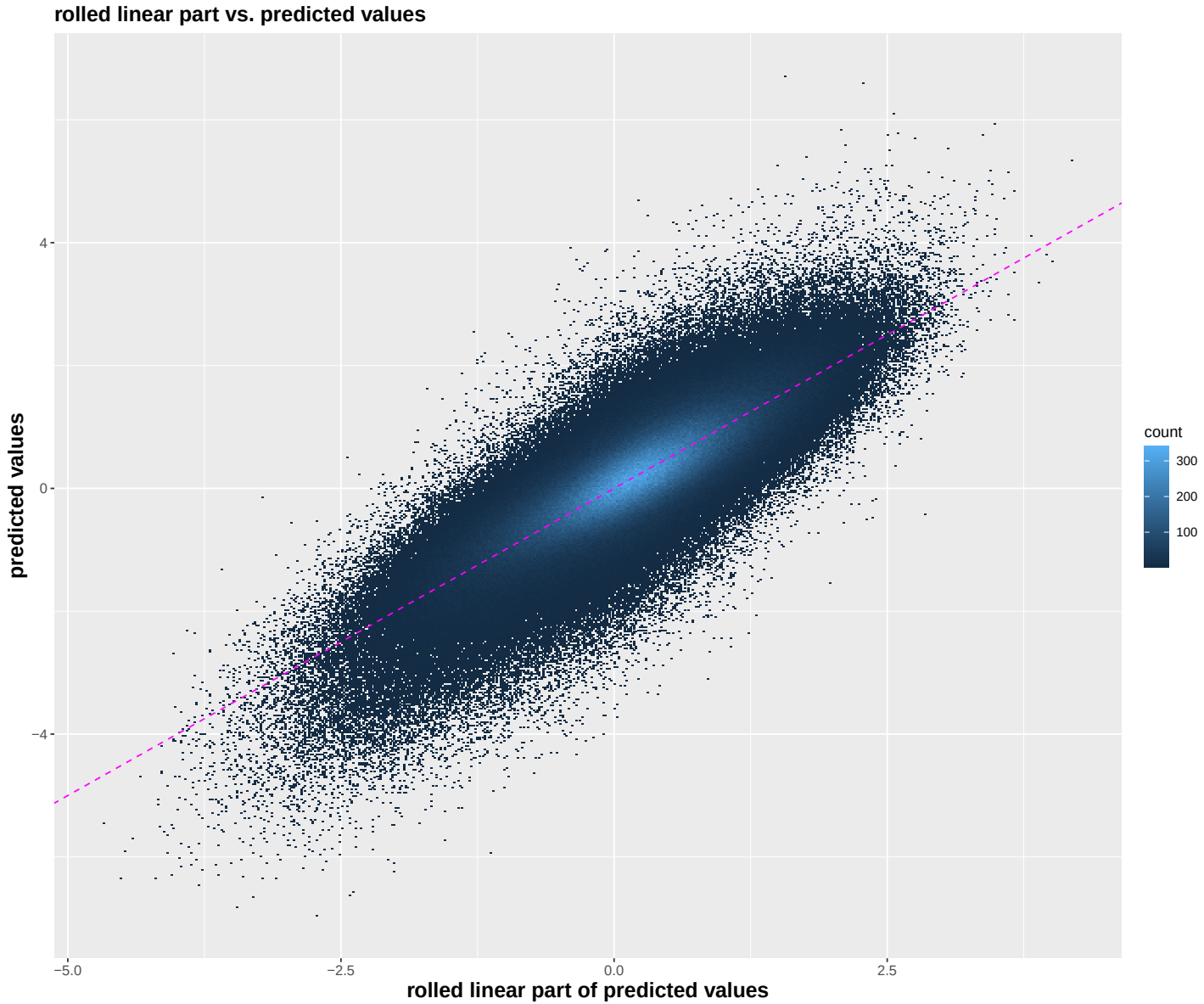
Then the "linear part" is just the fits and the nonlinear part is just the resids.

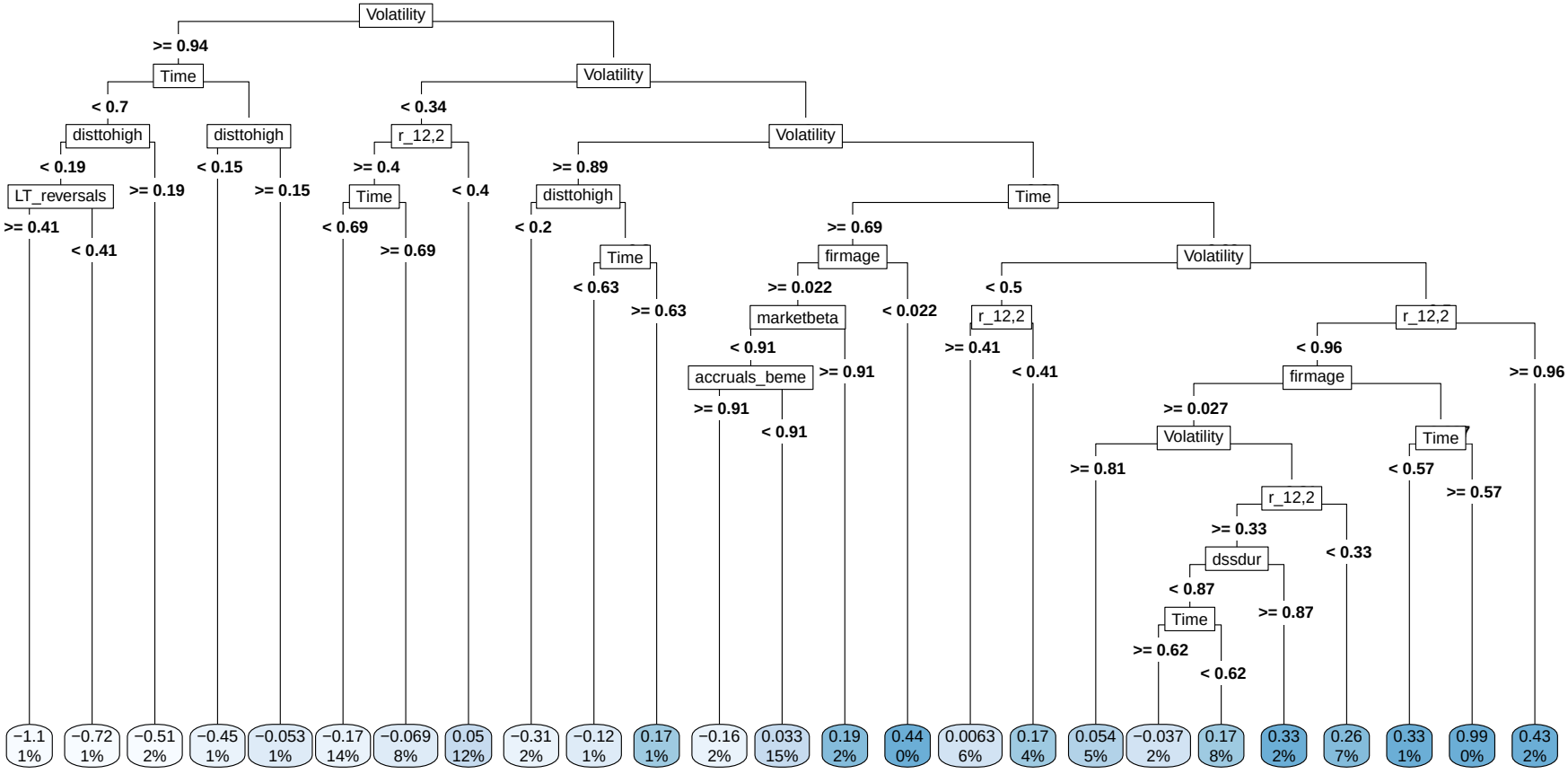
- each boxplots represent the linear coefficent over the 684 months
- remember regressions have y=predictions, not y = returns

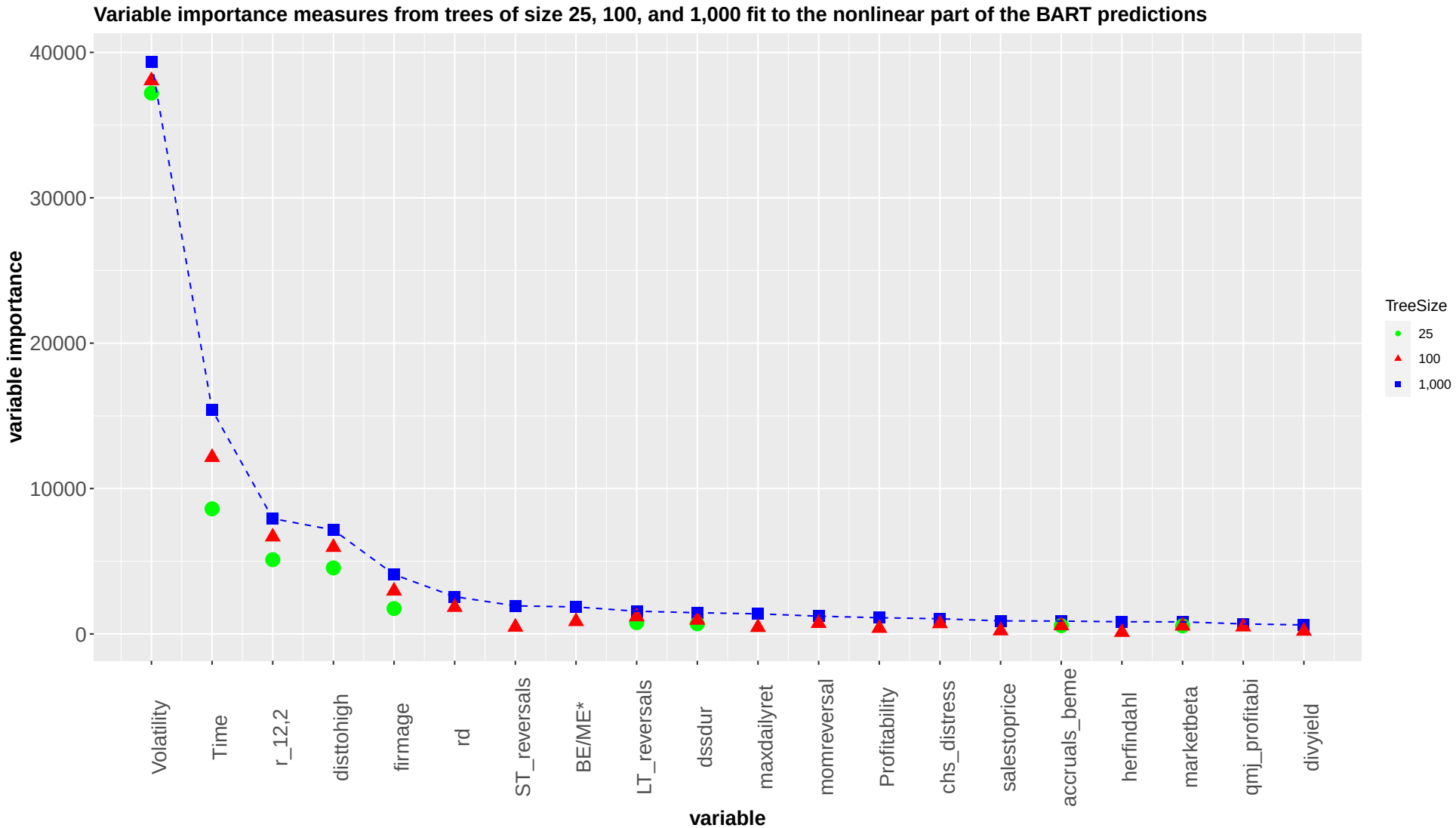


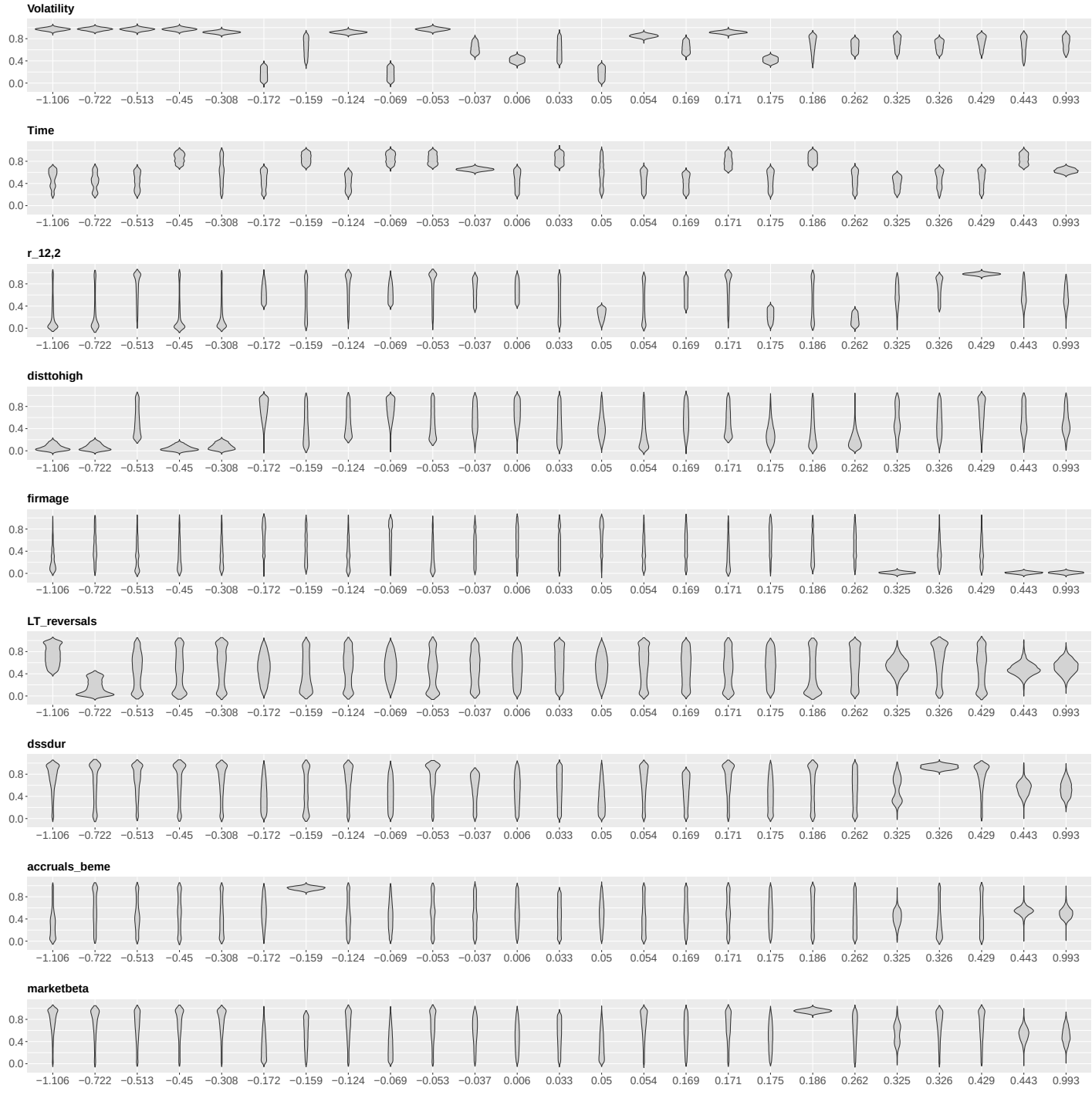


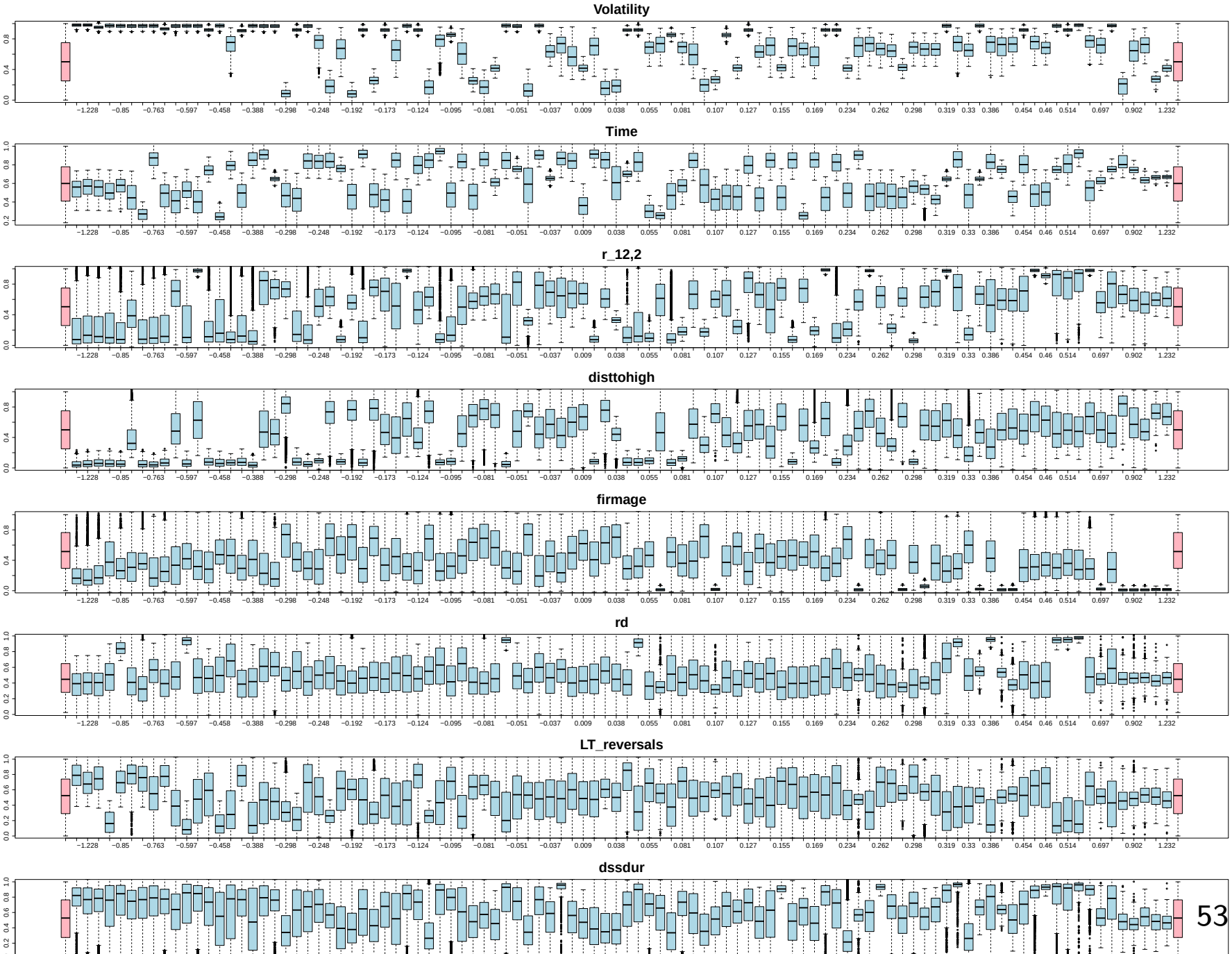

median of rolled regression coefficents, BART predictions on predictor variables, sorted by absolute value of coefficient
regressione coefficient
1
0
−1
ST_reversals
r_12,2
Volume
amihud
seasonality
Profitability
BE/ME*
Volatility
opleverage
salestoprice
divmonth
grossprofitab
chs_distress
earningsprice
Price
sharevolume
accruals_beme
maxdailyret
intmomentum
indadjcapxgro
ME
accruals
salesminusinv
ffprofitabili
assetgrowth
sustainablegr
netoperatinga
rd
marketbeta
BE/ME
oscore
divyield
profitability
returnonequit
disttohigh
seasonality_o
coskewness
I/A
salesgrowth
investmenttoc
netwcap_chang
advertising
inventorygrow
LT_reversals
aturnoverchg
firmage
herfindahl
profitmargin
leverage
momreversal
piotroskiF
tax
invgrowthrate
organizationc
debtissuance
qmj_profitabi
dssdur
zscore
cashflowtoequ
abnormalinves
total_xfin
EM
variable

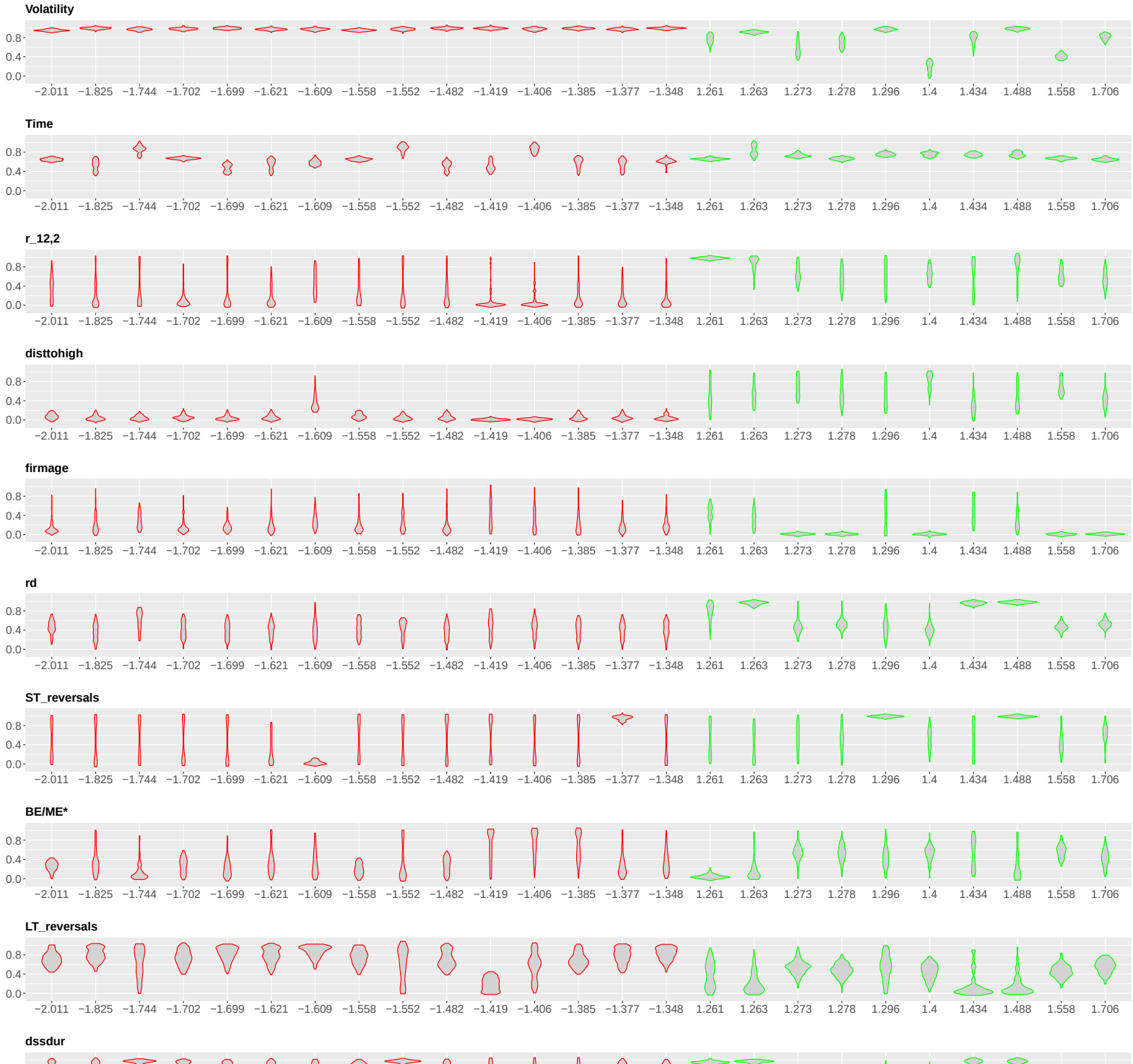

rolled linear part vs. predicted values
predicted values
4
0
−4
−5.0
−2.5
0.0
2.5
rolled linear part of predicted values
count
300
200
100


Volatility
>= 0.94
Time
< 0.7
disttohigh
< 0.19
LT_reversals
>= 0.41
< 0.41
>= 0.19
disttohigh
< 0.15
>= 0.15
Volatility
< 0.34
r_12,2
>= 0.4
Time
< 0.69
>= 0.69
< 0.4
Volatility
>= 0.89
disttohigh
< 0.2
Time
< 0.63
>= 0.63
Time
>= 0.69
firmage
>= 0.022
marketbeta
< 0.91
accruals_beme
>= 0.91
< 0.91
>= 0.91
< 0.022
Volatility
< 0.5
r_12,2
>= 0.41
< 0.41
r_12,2
< 0.96
firmage
>= 0.027
Volatility
>= 0.81
r_12,2
>= 0.33
dssdur
< 0.87
Time
>= 0.62
< 0.62
>= 0.87
< 0.33
Time
< 0.57
>= 0.57
>= 0.96
−1.1 1%
−0.72 1%
−0.51 2%
−0.45 1%
−0.053 1%
−0.17 14%
−0.069 8%
0.05 12%
−0.31 2%
−0.12 1%
0.17 1%
−0.16 2%
0.033 15%
0.19 2%
0.44 0%
0.0063 6%
0.17 4%
0.054 5%
−0.037 2%
0.17 8%
0.33 2%
0.26 7%
0.33 1%
0.99 0%
0.43 2%


Variable importance measures from trees of size 25, 100, and 1,000 fit to the nonlinear part of the BART predictions
variable importance
40000
30000
20000
10000
0
Volatility
Time
r_12,2
disttohigh
firmage
rd
ST_reversals
BE/ME*
LT_reversals
dssdur
maxdailyret
momreversal
Profitability
chs_distress
salestoprice
accruals_beme
herfindahl
marketbeta
qmj_profitabi
divyield
variable
TreeSize
25
100
1,000


Volatility
Time
r_12,2
disttohigh
firmage
LT_reversals
dssdur
accruals_beme
marketbeta
0.8
0.4
0.0
−1.106 −0.722 −0.513 −0.45 −0.308 −0.172 −0.159 −0.124 −0.069 −0.053 −0.037 0.006 0.033 0.05 0.054 0.169 0.171 0.175 0.186 0.262 0.325 0.326 0.429 0.443 0.993


Volatility
Time
r_12,2
disttohigh
firmage
rd
LT_reversals
dssdur
−1.228 −0.85 −0.763 −0.597 −0.458 −0.388 −0.298 −0.248 −0.192 −0.173 −0.124 −0.095 −0.081 −0.051 −0.037 0.009 0.038 0.055 0.081 0.107 0.127 0.155 0.169 0.234 0.262 0.298 0.319 0.33 0.386 0.454 0.46 0.514 0.697 0.902 1.232


Volatility
Time
r_12,2
disttohigh
firmage
rd
ST_reversals
BE/ME*
LT_reversals
dssdur
0.0
0.4
0.8
−2.011
−1.825
−1.744
−1.702
−1.699
−1.621
−1.609
−1.558
−1.552
−1.482
−1.419
−1.406
−1.385
−1.377
−1.348
1.261
1.263
1.273
1.278
1.296
1.4
1.434
1.488
1.558
1.706

7. Seeing the non-GAM part with T

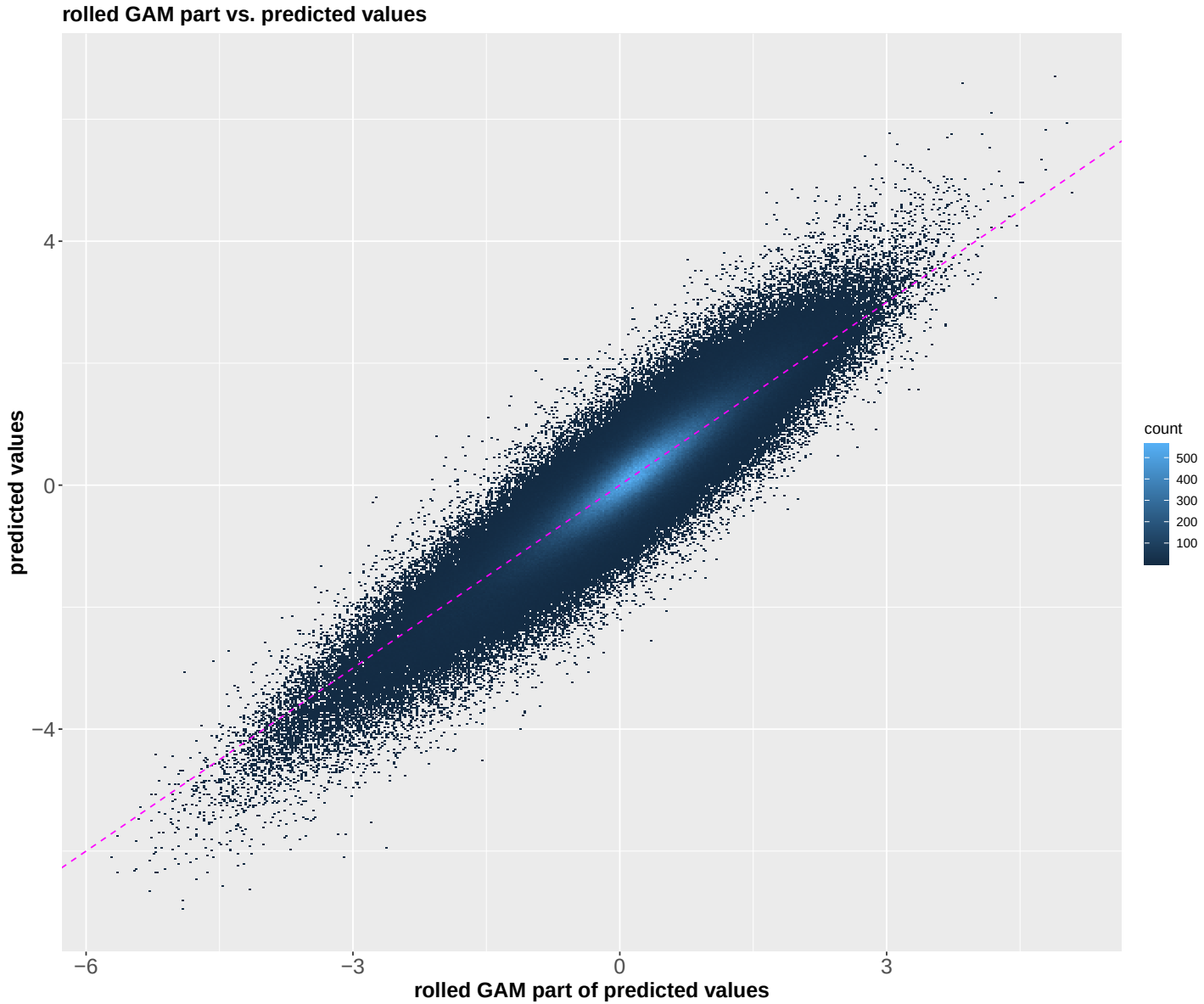
GAM: $f(x_1, x_2, \ldots, x_p) = \sum_{i=1}^{p} f_i(x_i)$.

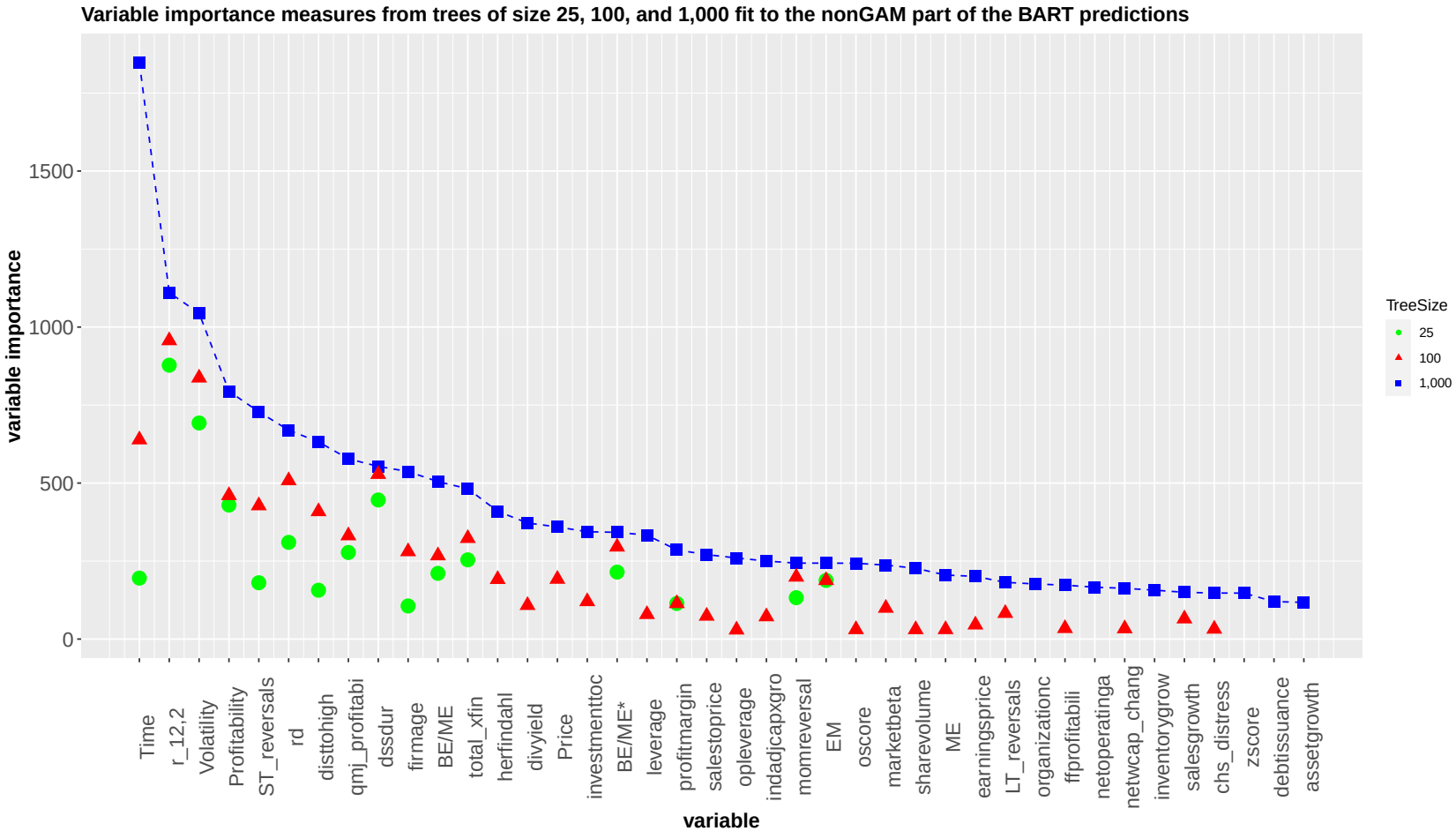
The idea of *interactions* is an evocative theme in statistics.

Let's fit trees to the "non-GAM" part of the predictions to zero in on the interactions !!

Let's look at the version where we fit a GAM to the BART predictions each month and get the fits and resids.
The stacked resids will be our dependent variable in the tree fits and we will include T in the regressors.

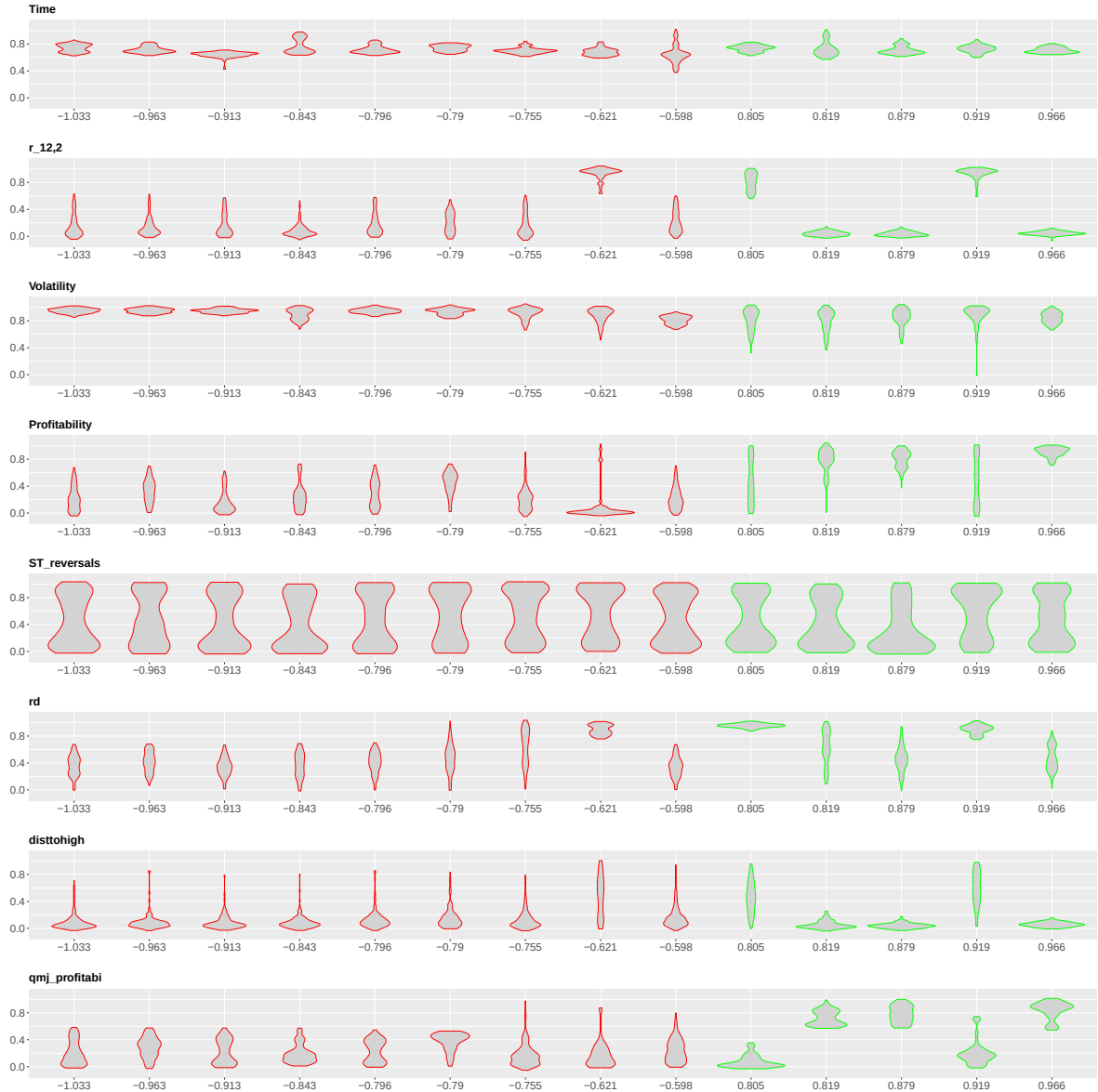
The simpler version does not include T in the tree fit and just fits one overall GAM to the predictions.
That is what I showed you before.


rolled GAM part vs. predicted values
predicted values
4
0
−4
−6
−3
0
3
rolled GAM part of predicted values
count
500
400
300
200
100



It makes sense that the smaller trees fail to capture complex interactions !!!

Tree size 1,000.



8. Conclusion

I wish I had a dollar for every plot I have made.

I wish I had a dollar for every plot I am going to make.

Hard to really understand time-varying predictions with 62 variables.

But, with "simple" graphics, some basic things are made quite clear.

a quote from Gu, Kelly, Xiu:

```
"The most successful predictors are
   price trends, liquidity, and volatility."
```

So, big picture we agree with Gu et. al. but add a few more.

Nice confirmation since much of what we done is different *and* we have much more of a feeling for what kinds or roles the key variables play.