Evaluate testing data (regression) - xgboost

EVE W.

2019-11-16

Contents


```
## user input
project_home <- "~/EVE/examples"
project_name <- "xgboostR_regression_1"
```

0. Load Data

```
## 300 of samples were used
```

```
## 101 of full features
```

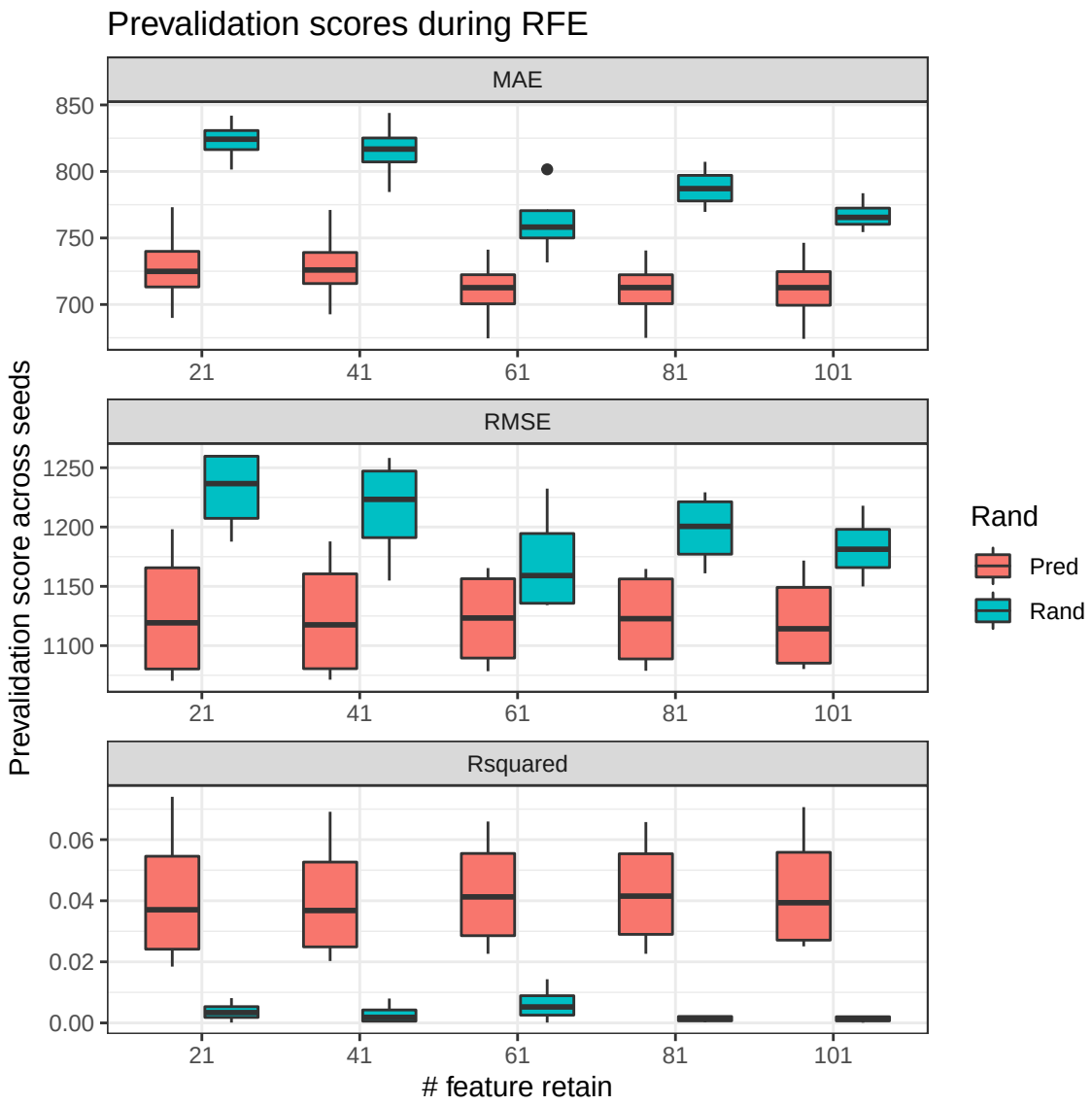
```
## 4 runs, each run contains 3 CVs.
```

```
## os_time :
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   182.8   480.0   889.4  1221.2  7125.0
```

run with XGBoost.r with evaluation metric: default.

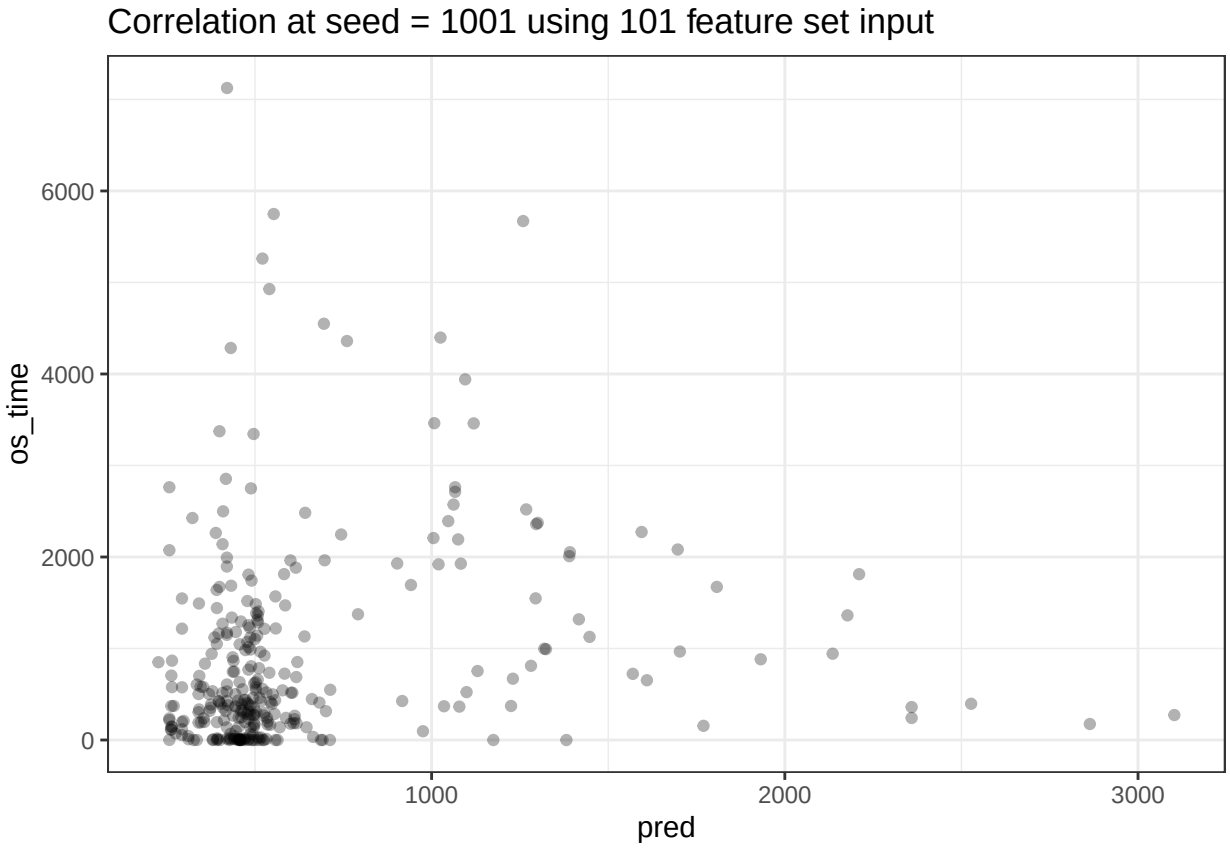
1. Scores



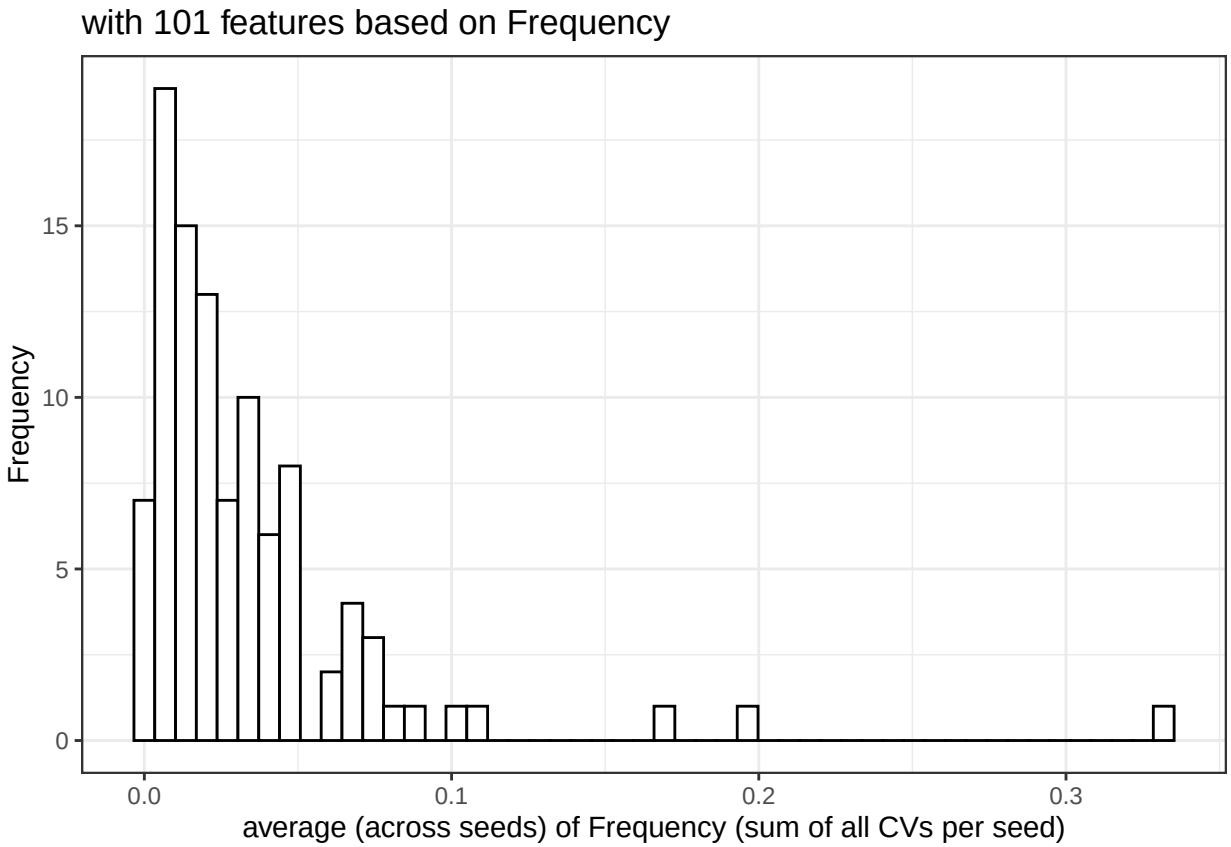
metrics	size.max	median.max	size.min	median.min
MAE	41	725.942	61	712.645
RMSE	61	1123.314	101	1114.231
Rsquared	81	0.042	41	0.037

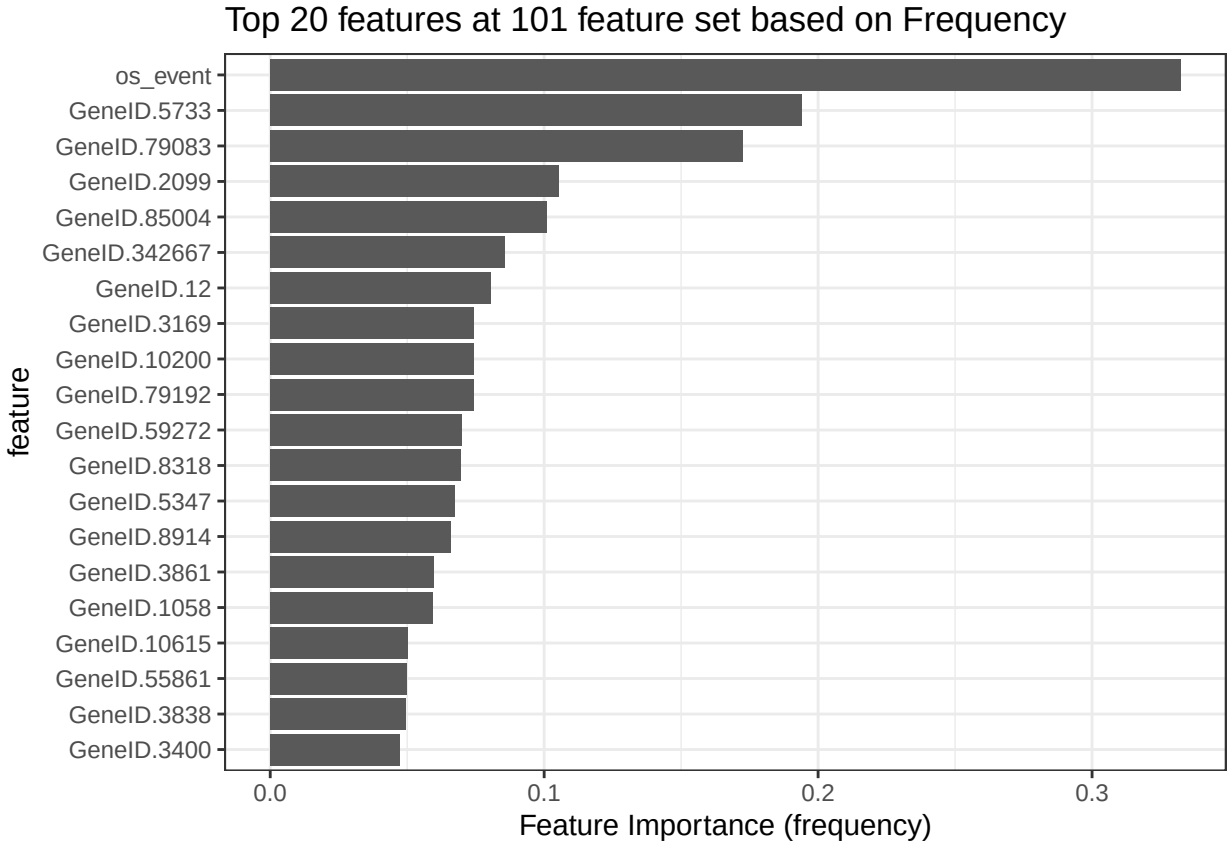
correlation

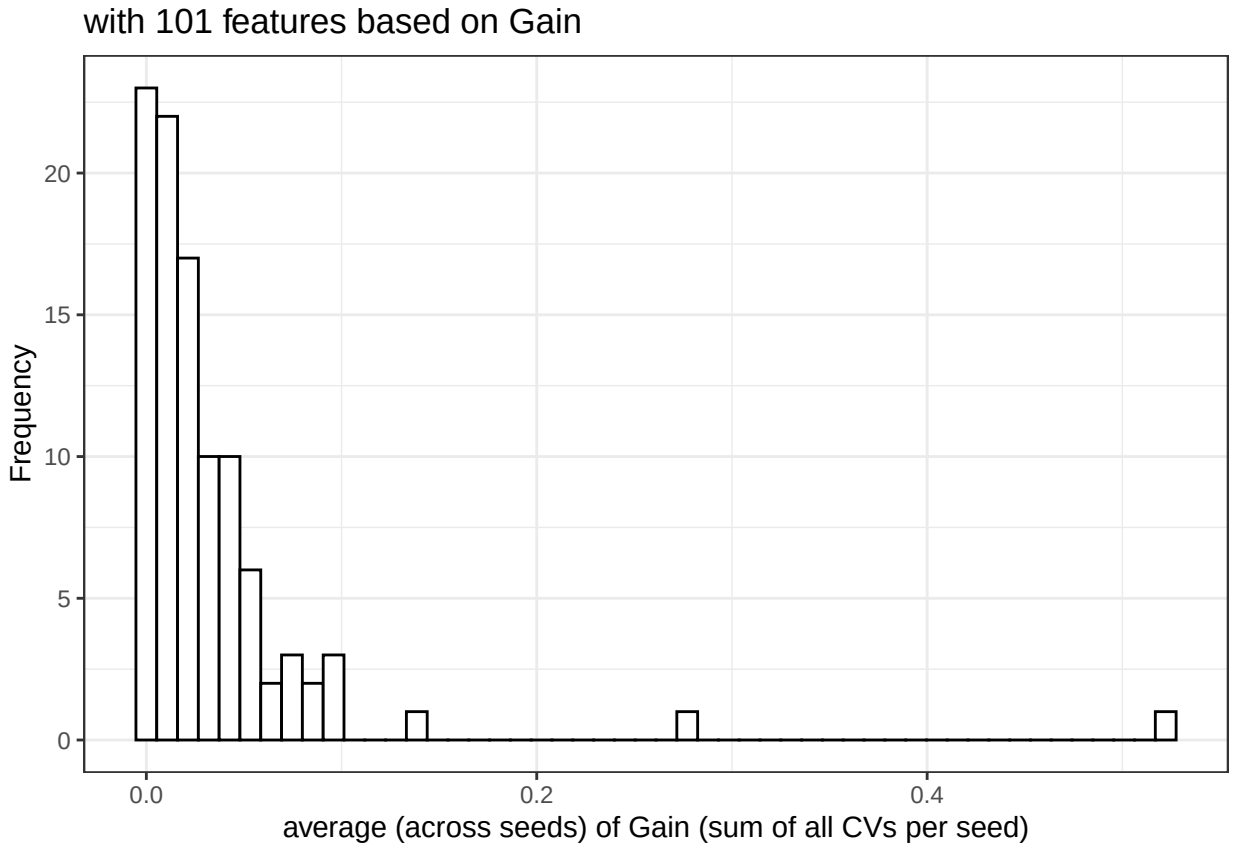
```
##
##
## Table: Averaged pearson correlation across seeds
##
##    cor.avg     cor.sdt
## ----------  ----------
##  0.1980039   0.0505366
```

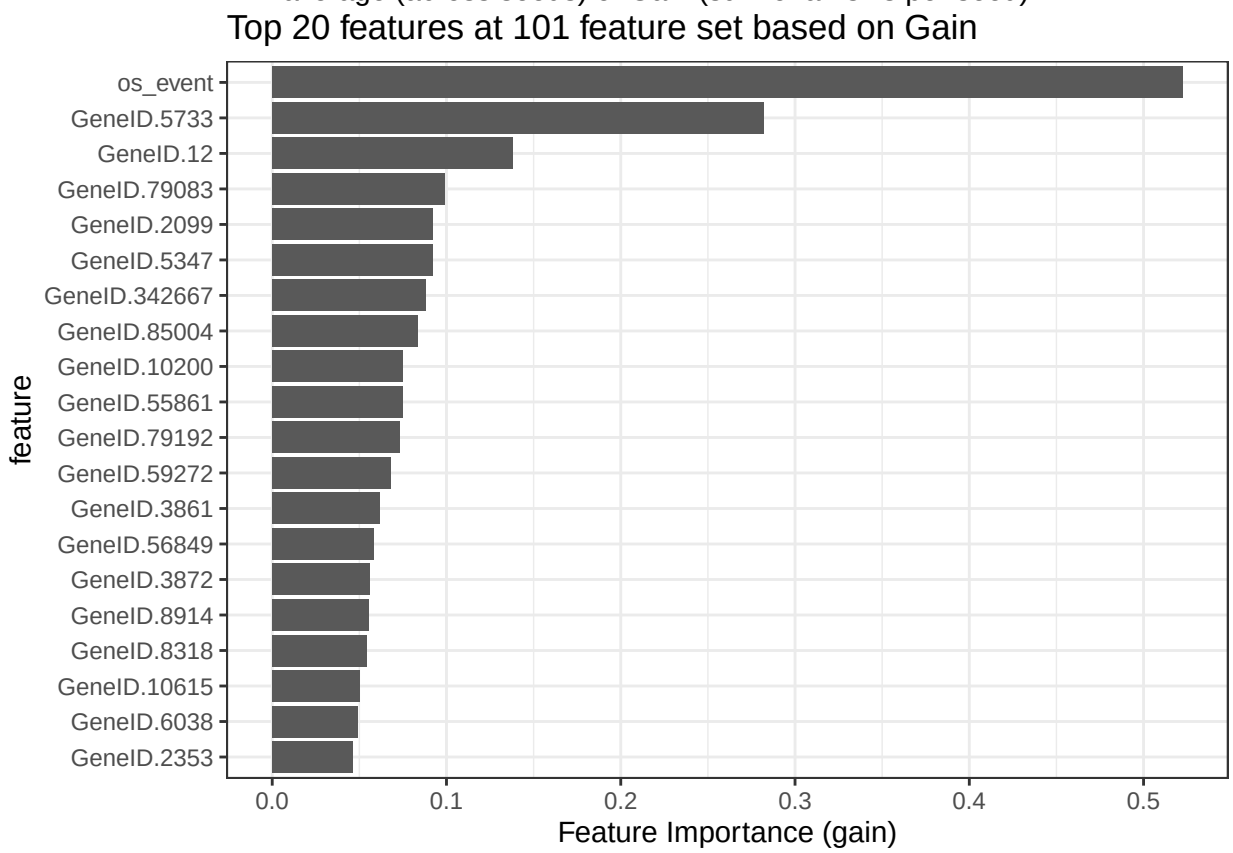

Correlation at seed = 1001 using 101 feature set input
os_time
6000
4000
2000
0
1000
2000
3000
pred

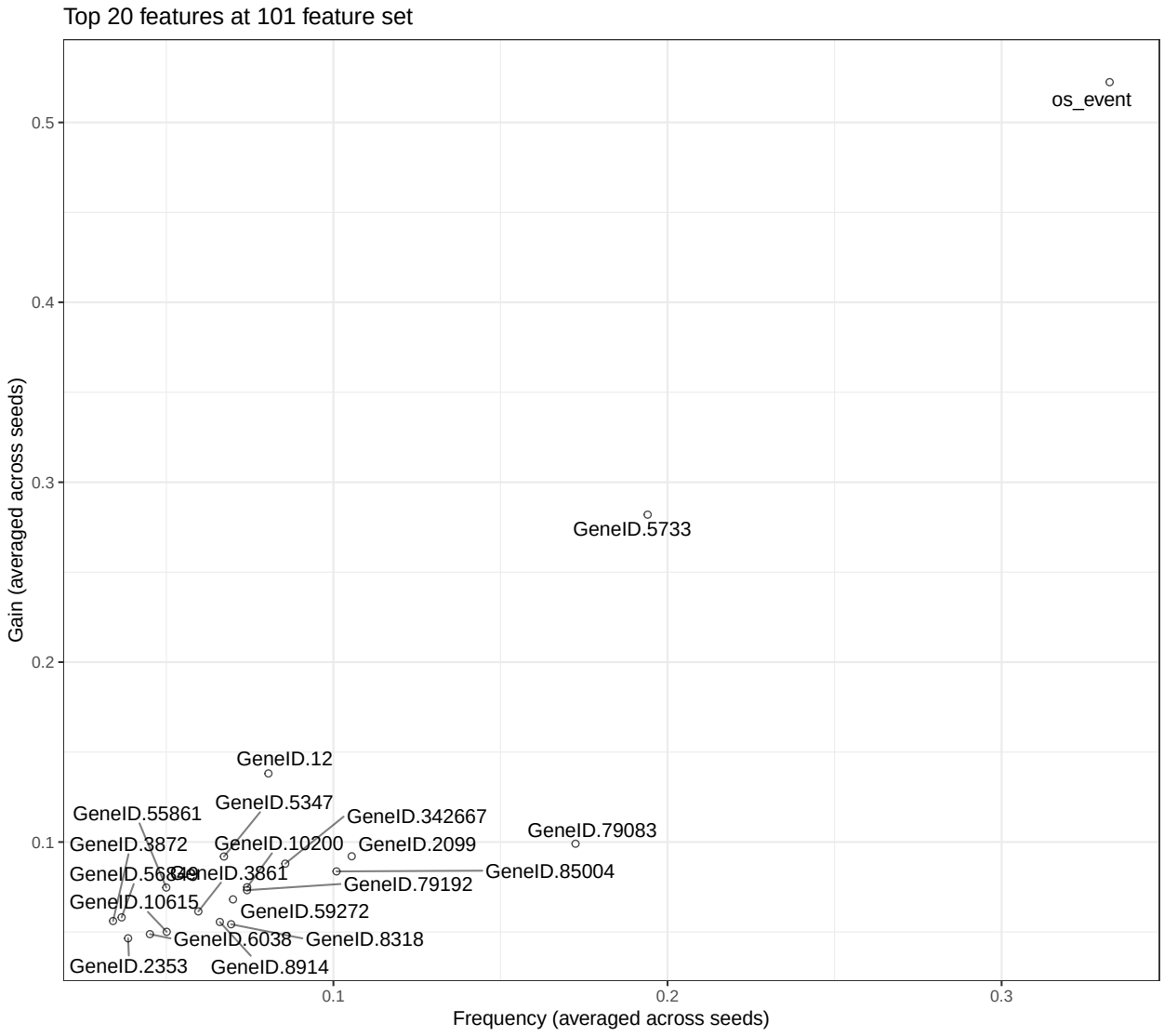
2. Important Features






with 101 features based on Gain
Frequency
20
15
10
5
0
0.0
0.2
0.4
average (across seeds) of Gain (sum of all CVs per seed)


Top 20 features at 101 feature set based on Gain
feature
os_event
GeneID.5733
GeneID.12
GeneID.79083
GeneID.2099
GeneID.5347
GeneID.342667
GeneID.85004
GeneID.10200
GeneID.55861
GeneID.79192
GeneID.59272
GeneID.3861
GeneID.56849
GeneID.3872
GeneID.8914
GeneID.8318
GeneID.10615
GeneID.6038
GeneID.2353
0.0
0.1
0.2
0.3
0.4
0.5
Feature Importance (gain)

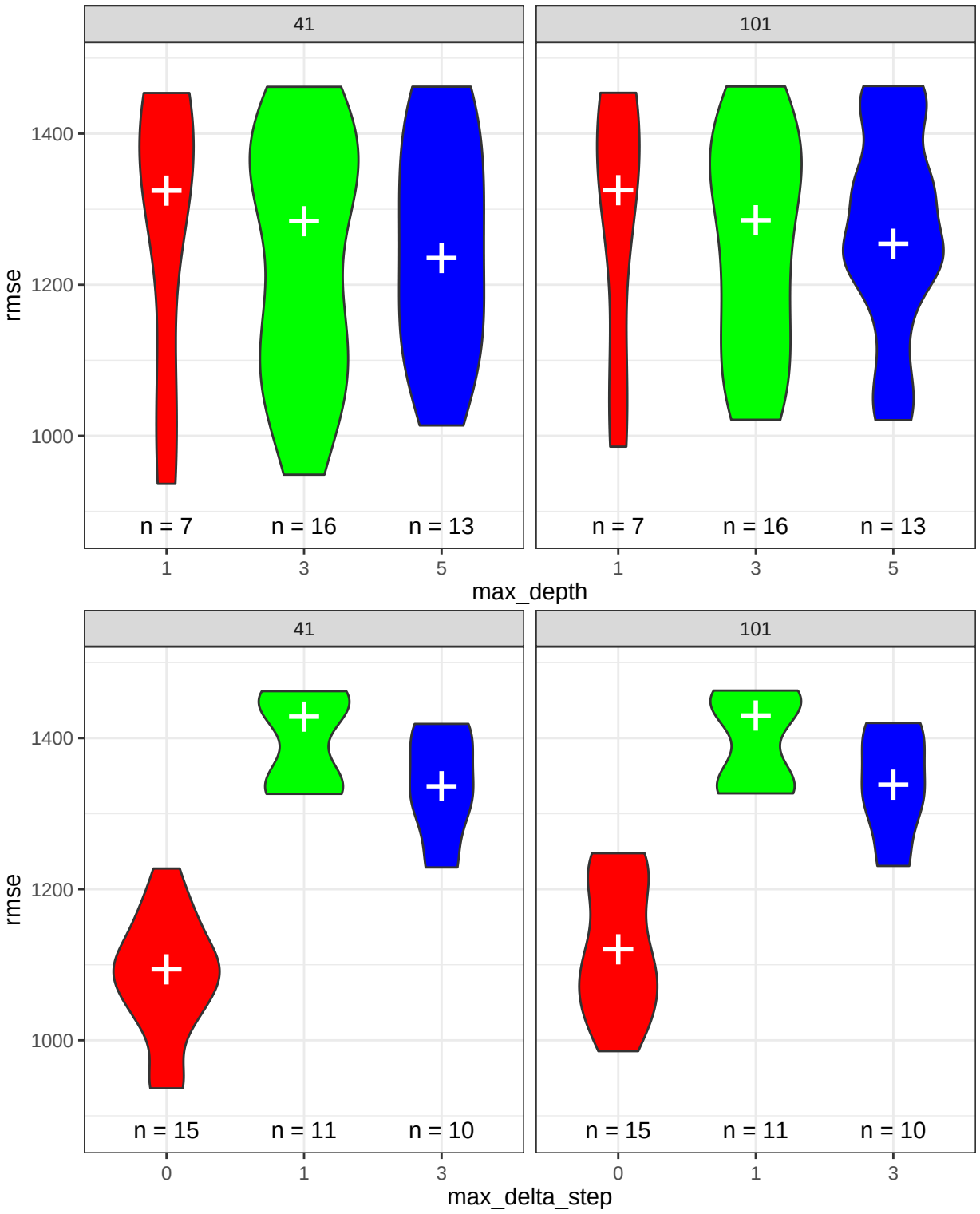


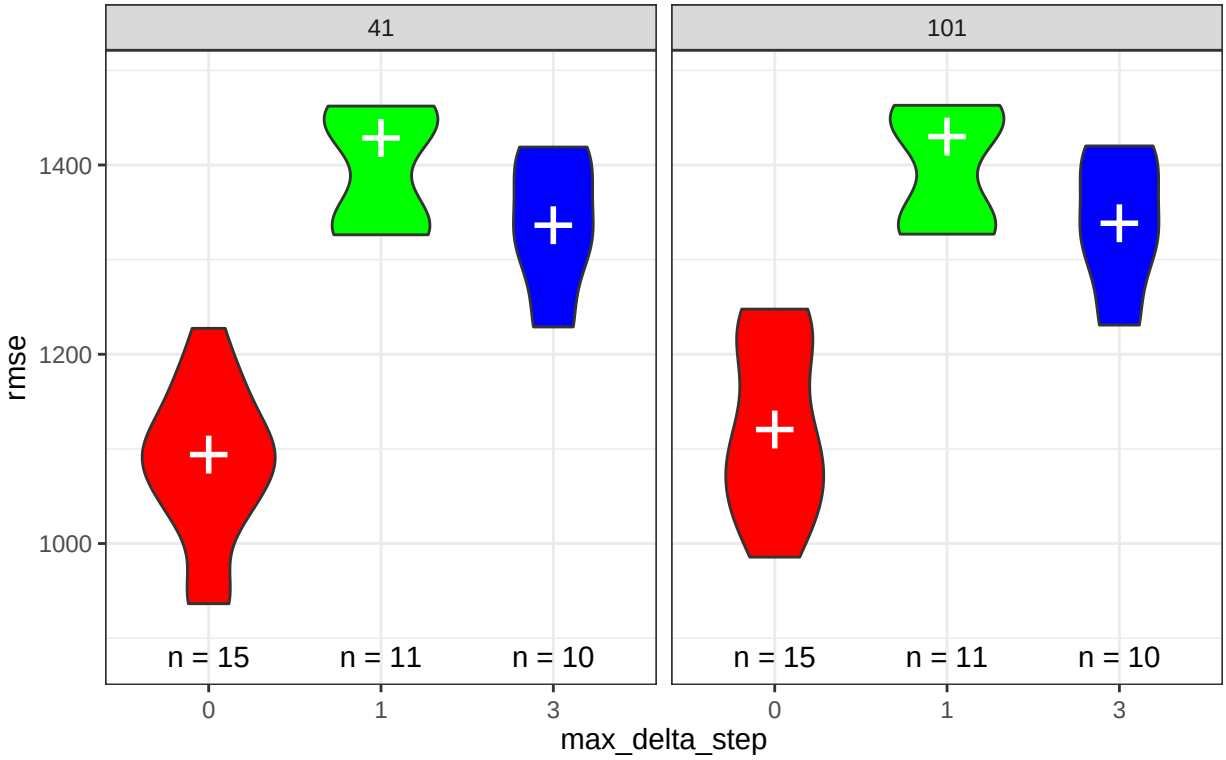
3. Hyper-parameters

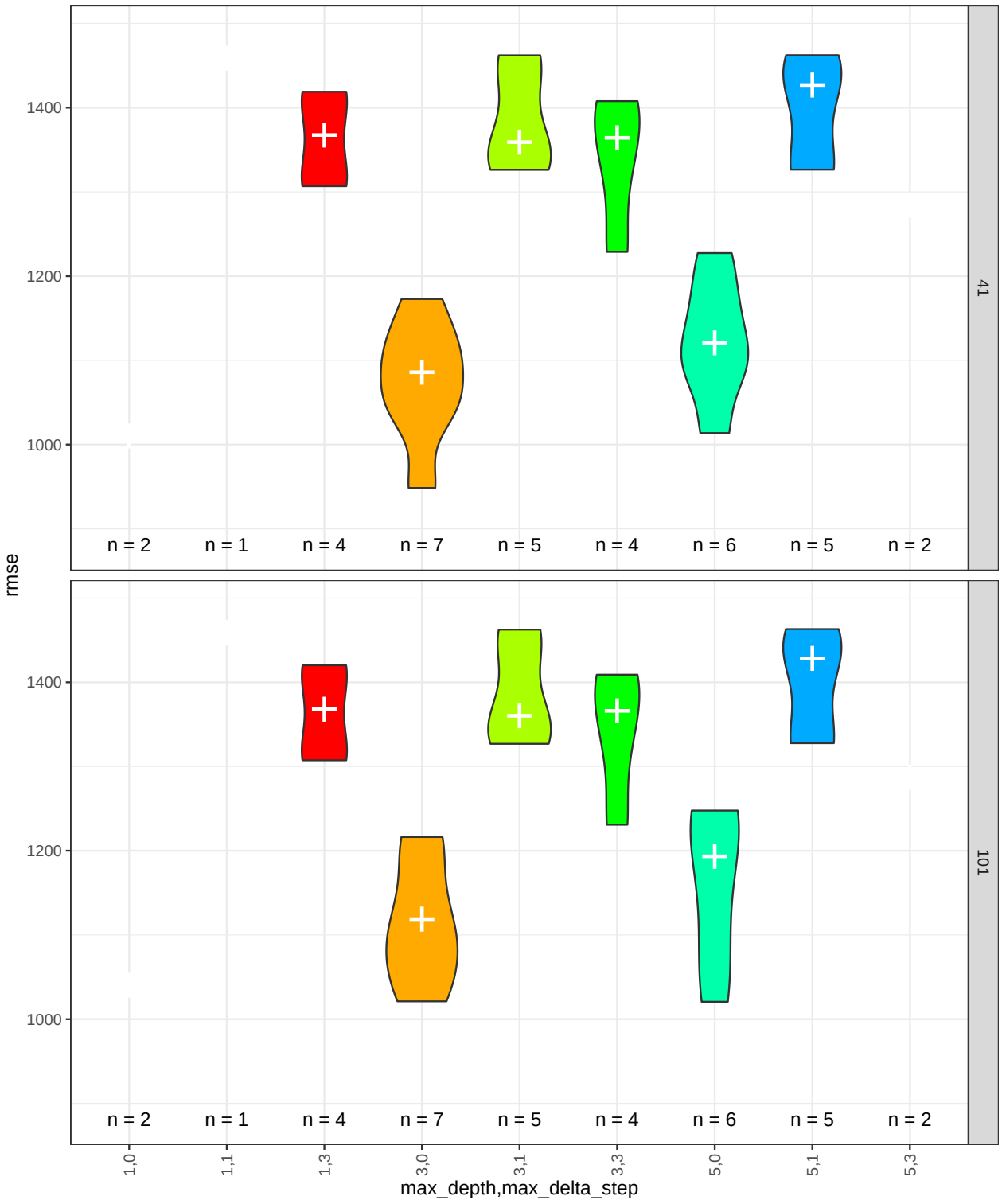
```
## Warning: `cols` is now required.
## Please use `cols = c(df)`
```

parameter optimization file (72 records) includes 4 seeds. Each seed generates 3 cv splits. Within each cv split, there is a 2 step RFE (at 41, 101). So 72 / 4 / 3 / 2 = 3 parameter combinations tried in each cv split.

all grid search results

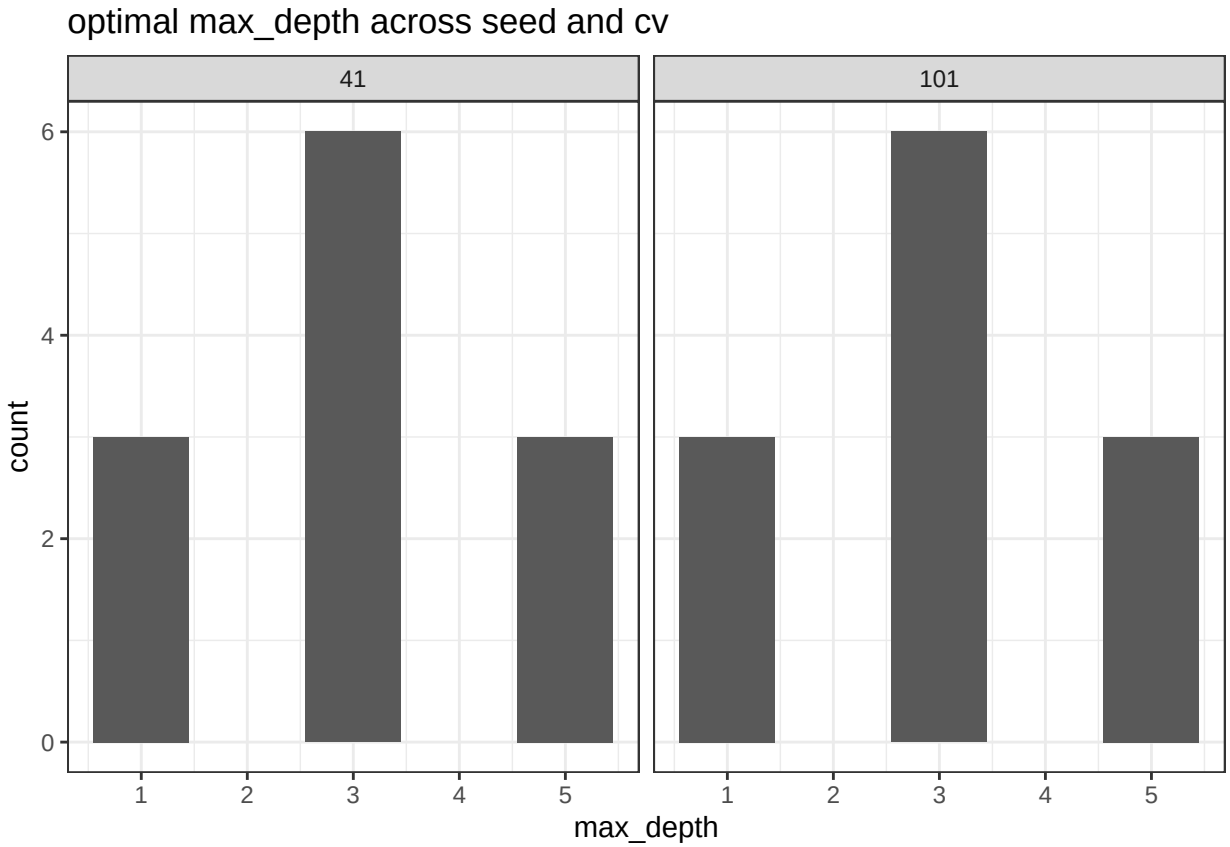


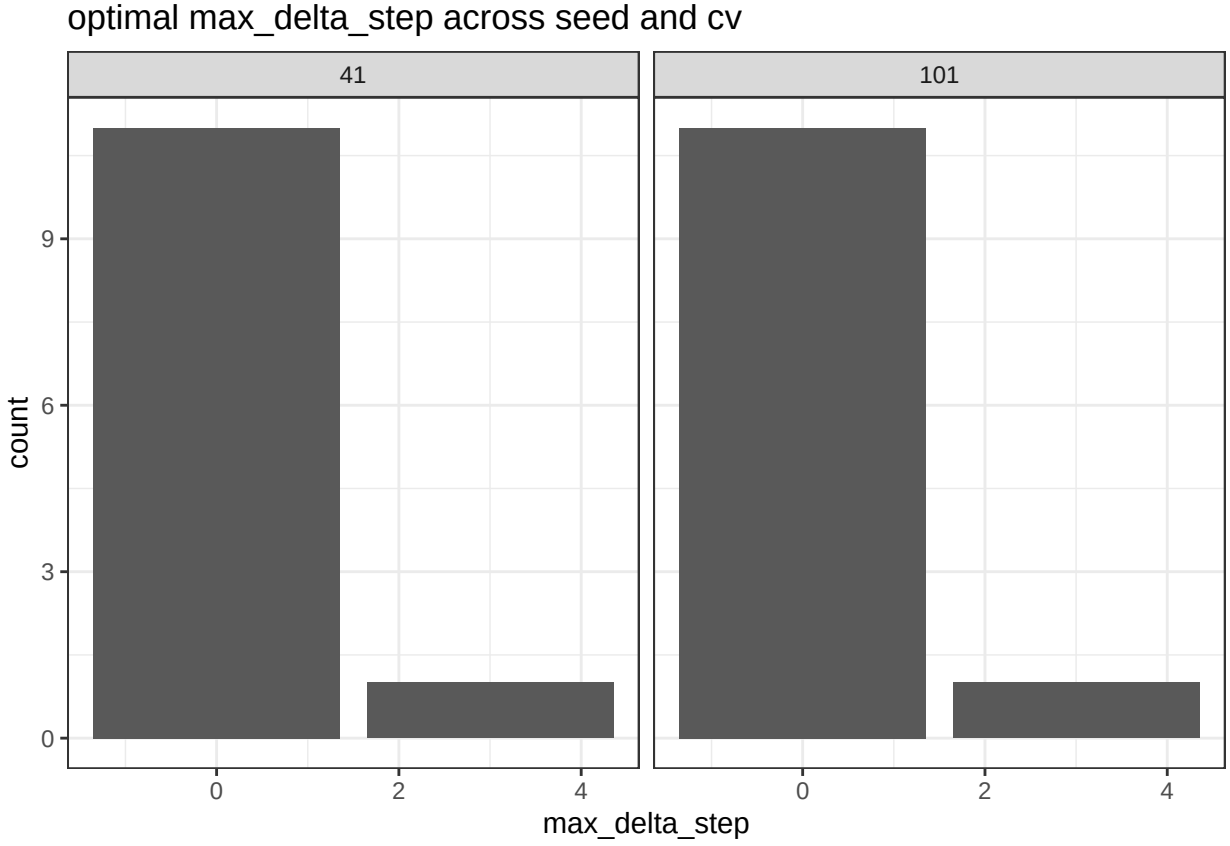

41
101
1400
1200
1000
rmse
n = 15
n = 11
n = 10
n = 15
n = 11
n = 10
0
1
3
0
1
3
max_delta_step

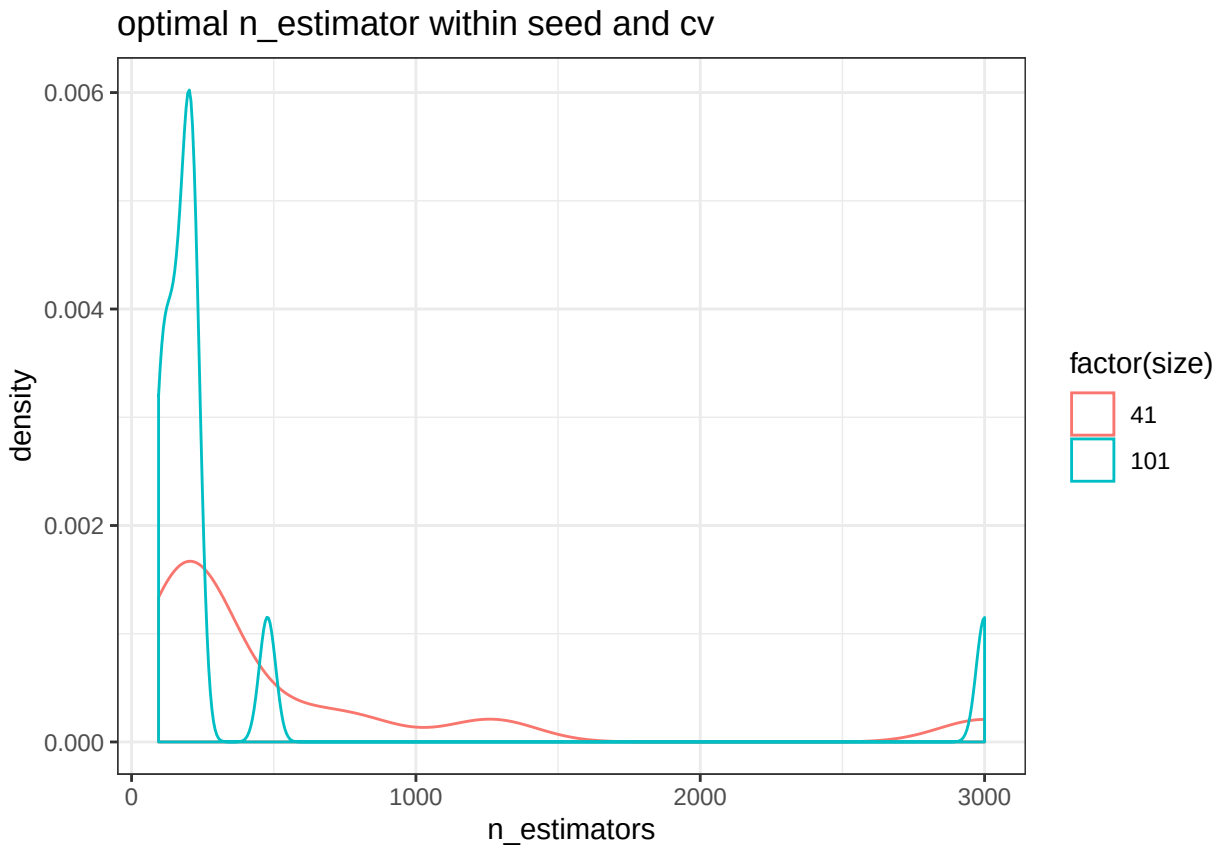


over best parameter combo per cv

Note the 2nd /3rd best parameter combinations might not be too bad either.


optimal max_depth across seed and cv
41
101
count
0
2
4
6
1
2
3
4
5
max_depth


optimal max_delta_step across seed and cv
41
101
count
0
3
6
9
0
2
4
max_delta_step



more about the best parameter combination selection

```
select_ft_step <- 101

df1 <- subset(grid_best, size==select_ft_step & max_depth==5 &  max_delta_step == 0 )
print( paste('summary of n estimator at',select_ft_step, 'feature step'))
```

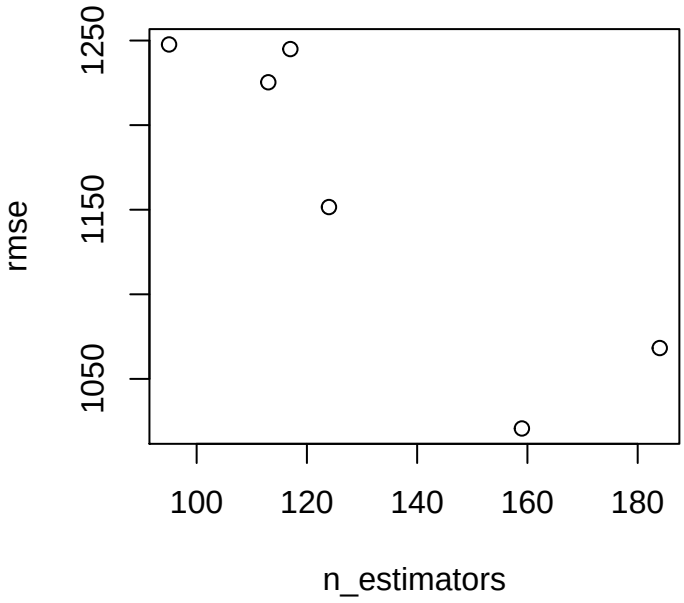
```
## [1] "summary of n estimator at 101 feature step"
```

```
print(summary(df1$n_estimators))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    95.0   109.5   124.0   126.0   141.5   159.0
```

```
df2 <- subset(df.grid, size==select_ft_step & max_depth==5 &  max_delta_step == 0 )

with(df2, plot(x = n_estimators, y=score, ylab=score_label))
```


rmse
1250
1150
1050
100
120
140
160
180
n_estimators