Evaluate testing data (binary-class) - XGBoost

EVE W.

2019-04-05

Contents

```
## user input
project_home <- "~/EVE/examples"
project_name <- "xgboostR_binary_1"
```

0. Load Data

```
## Warning: `data_frame()` is deprecated, use `tibble()`.
## This warning is displayed once per session.
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   Patient_ID = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
## 199 of samples were used
```

```
## 100 of full features
```

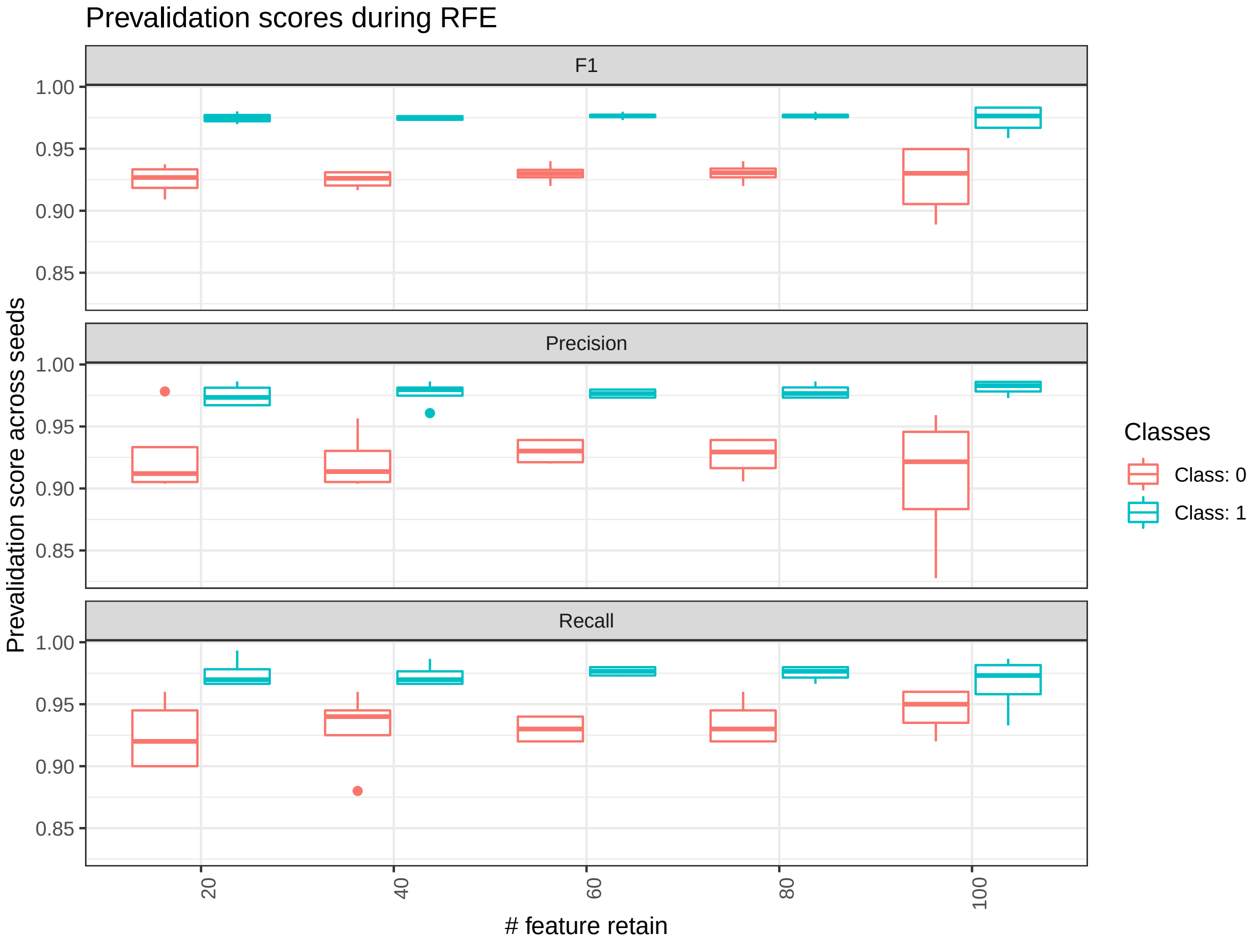
```
## 4 runs, each run contains 3 CVs.
```

```
## Labels:
```

run with XGBoost.r evaluation metric: f1_harmonic2.

1. Scores

1.1 Scores per Class



Confusion Matrix

```
## confusion matrix at feature size = 100
## sum across 4 seeds
```

```
##           Reference
## Prediction   0   1
##          0 189  20
##          1  11 576
```

1.2 Average score

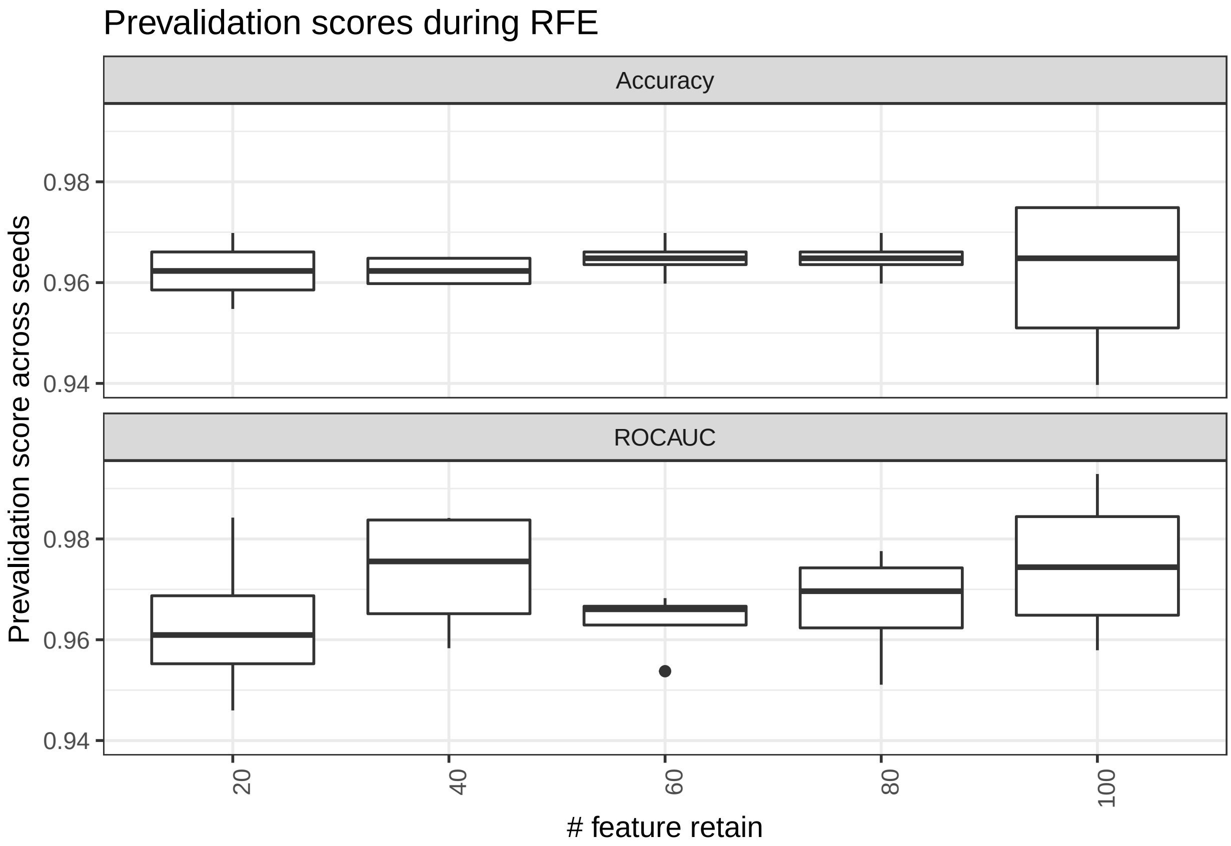
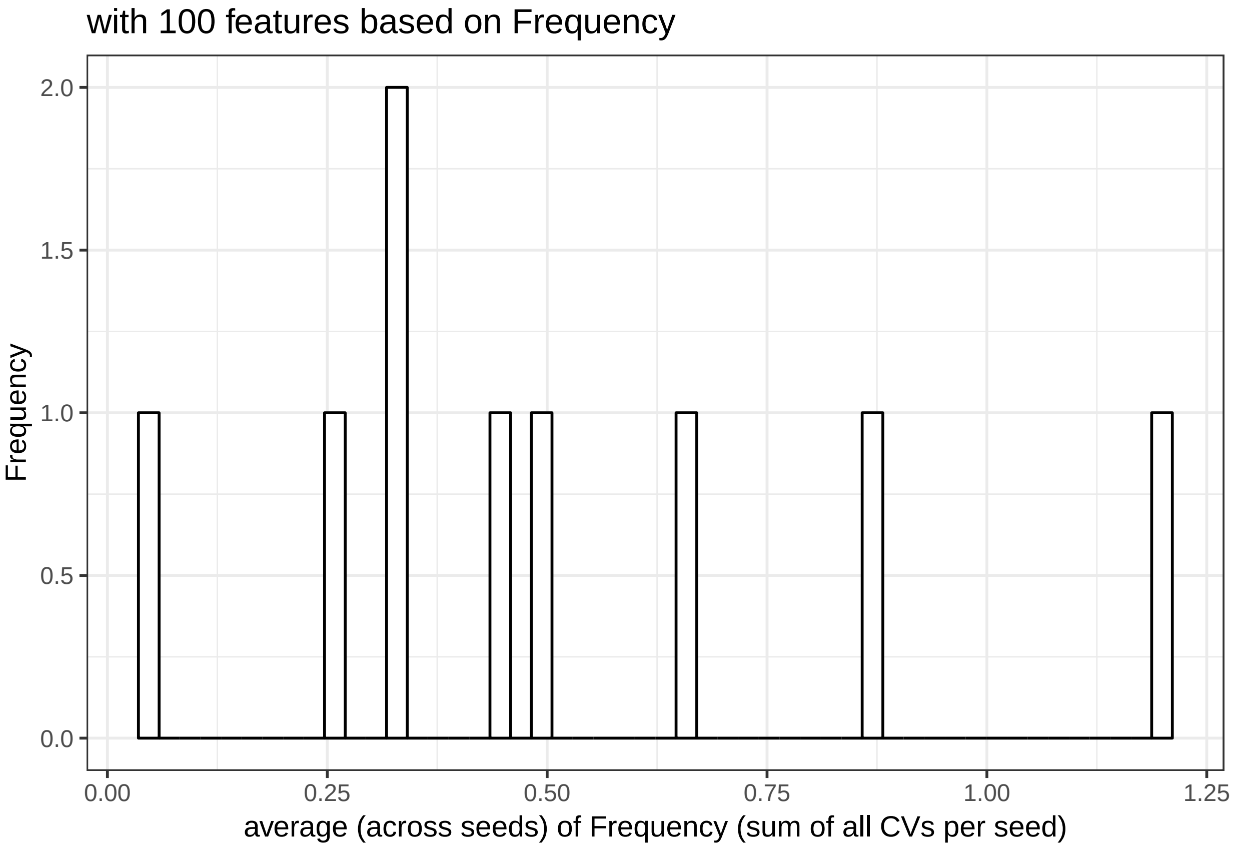
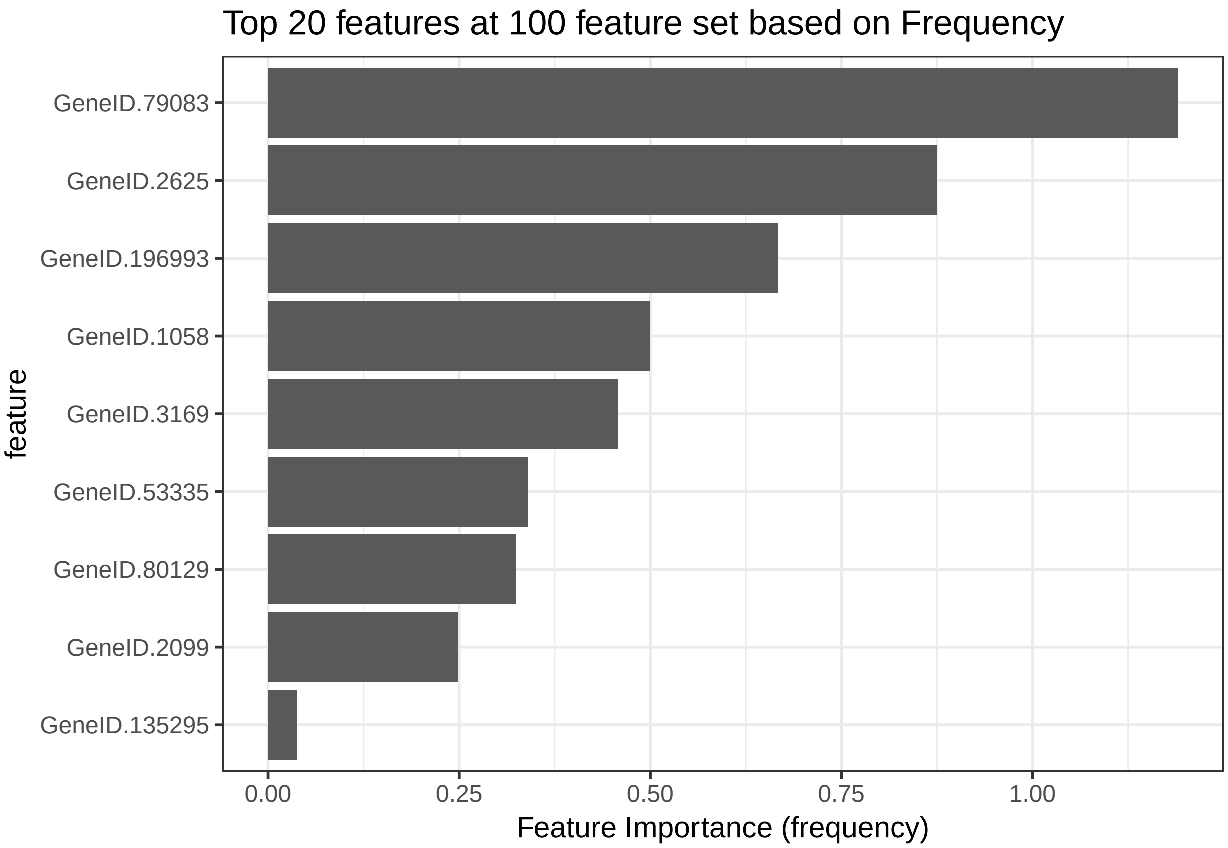


Table 1: best scores

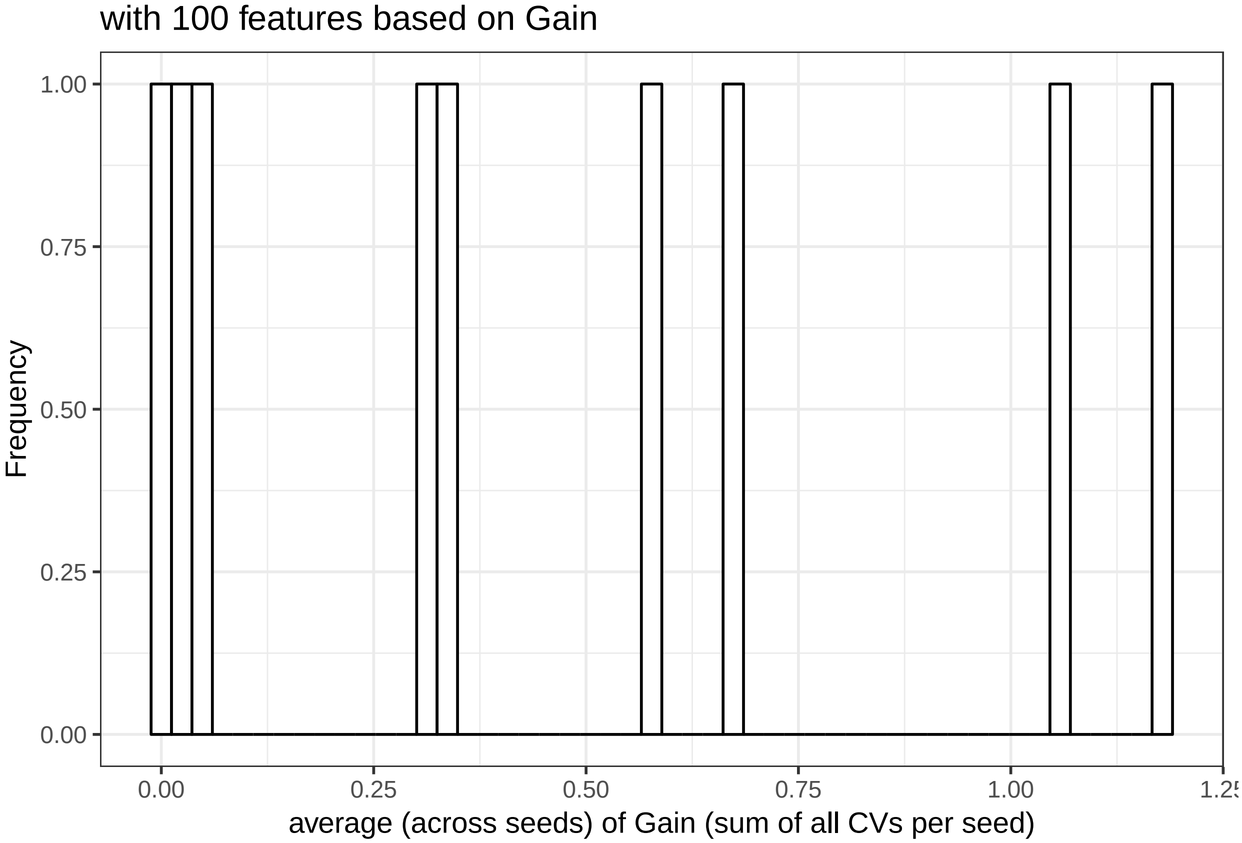
metrics	size.max	median.max	size.min	median.min
Accuracy	60	0.965	20	0.962
F1	80	0.954	40	0.950
Precision	60	0.953	20	0.944
Recall	80	0.955	20	0.950
ROCAUC	40	0.976	20	0.961

2. Important Features

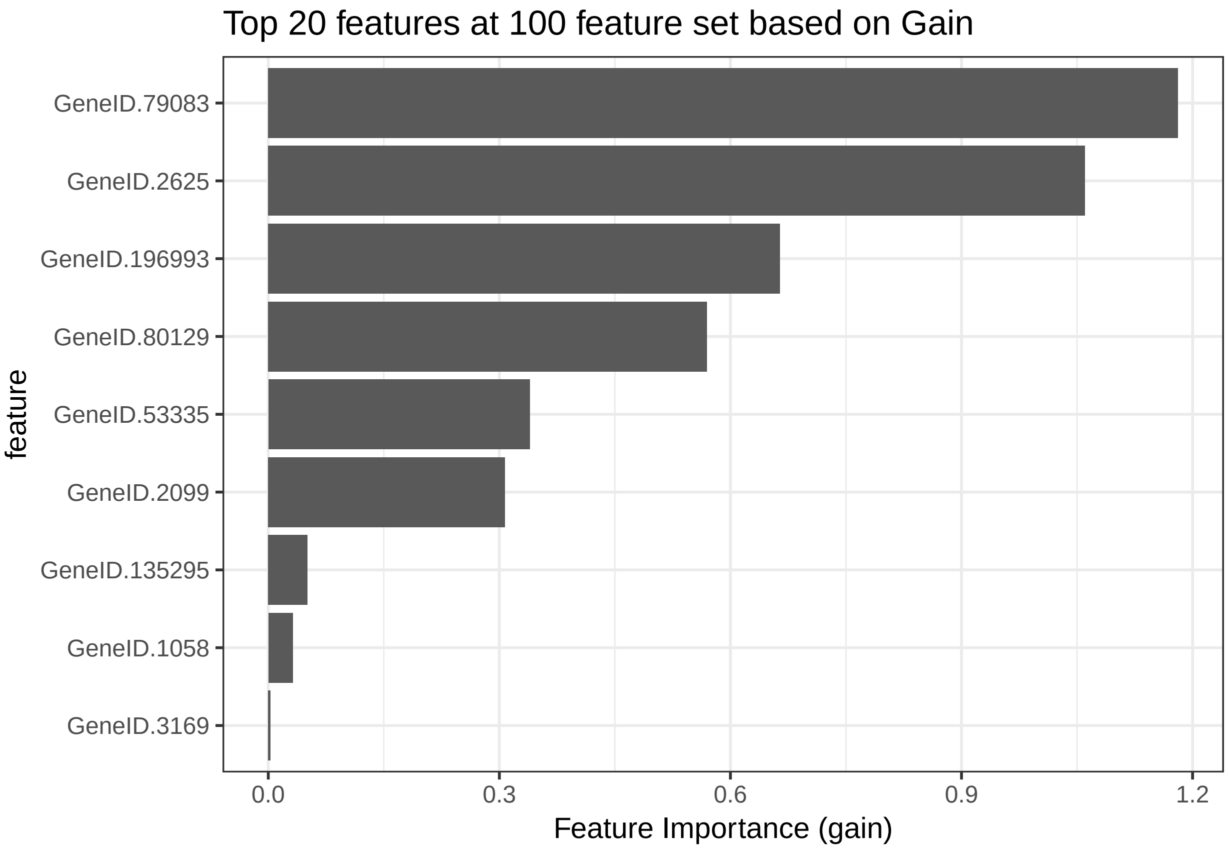


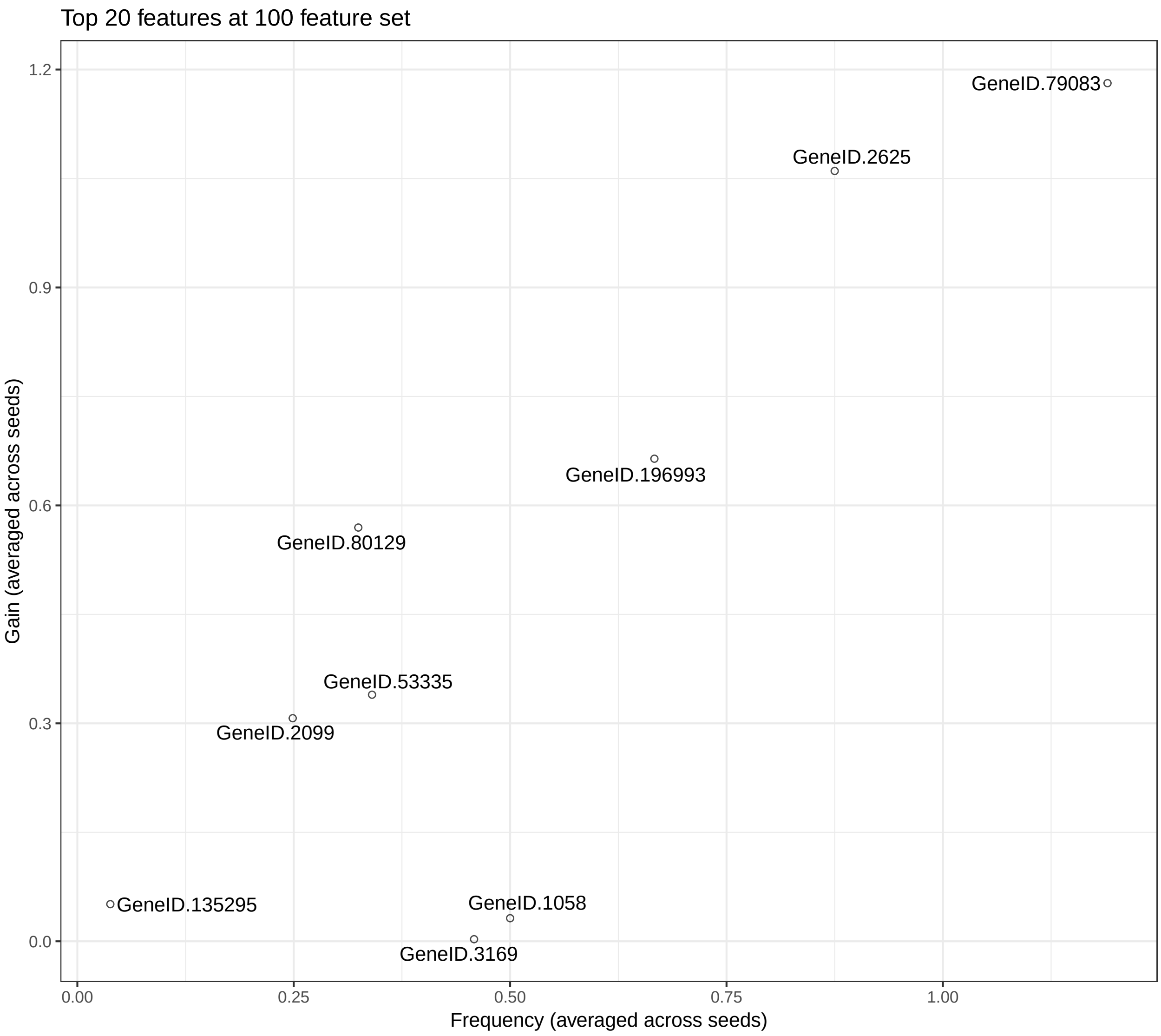


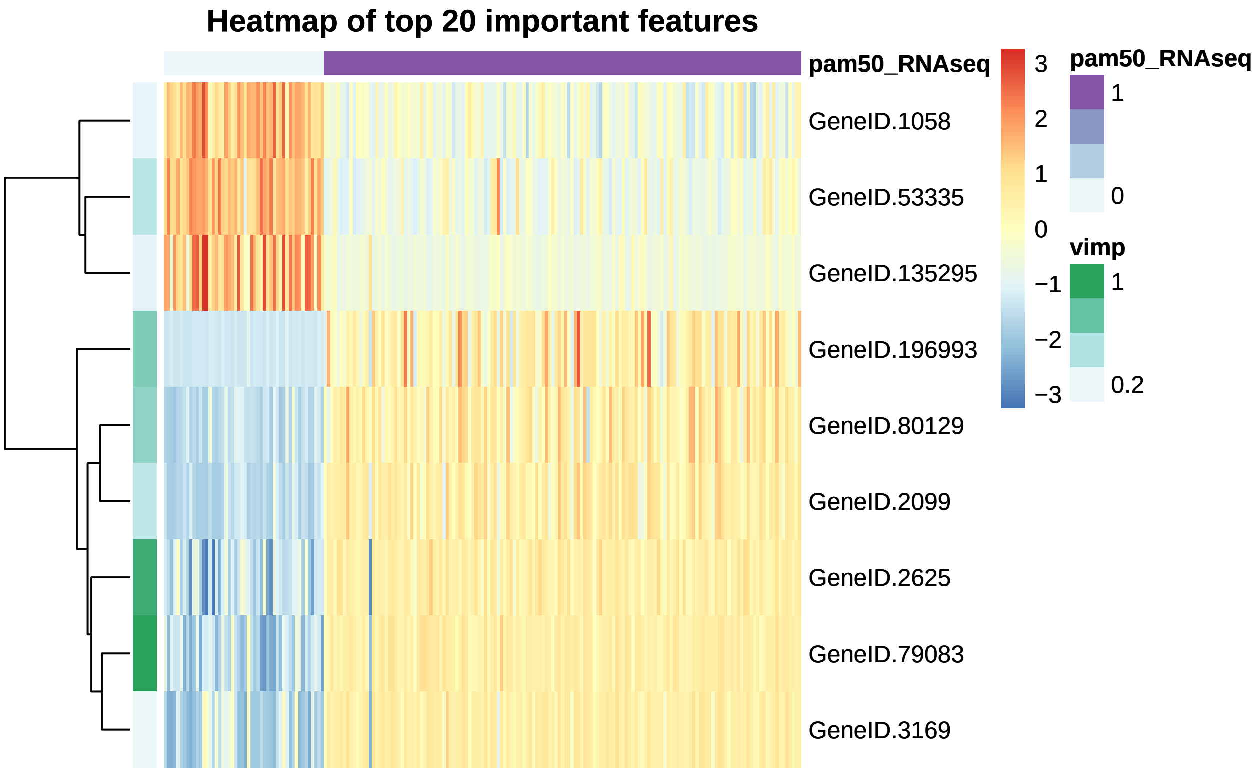
Top 20 features at 100 feature set based on Frequency



with 100 features based on Gain


Top 20 features at 100 feature set based on Gain
feature
GeneID.79083
GeneID.2625
GeneID.196993
GeneID.80129
GeneID.53335
GeneID.2099
GeneID.135295
GeneID.1058
GeneID.3169
0.0
0.3
0.6
0.9
1.2
Feature Importance (gain)

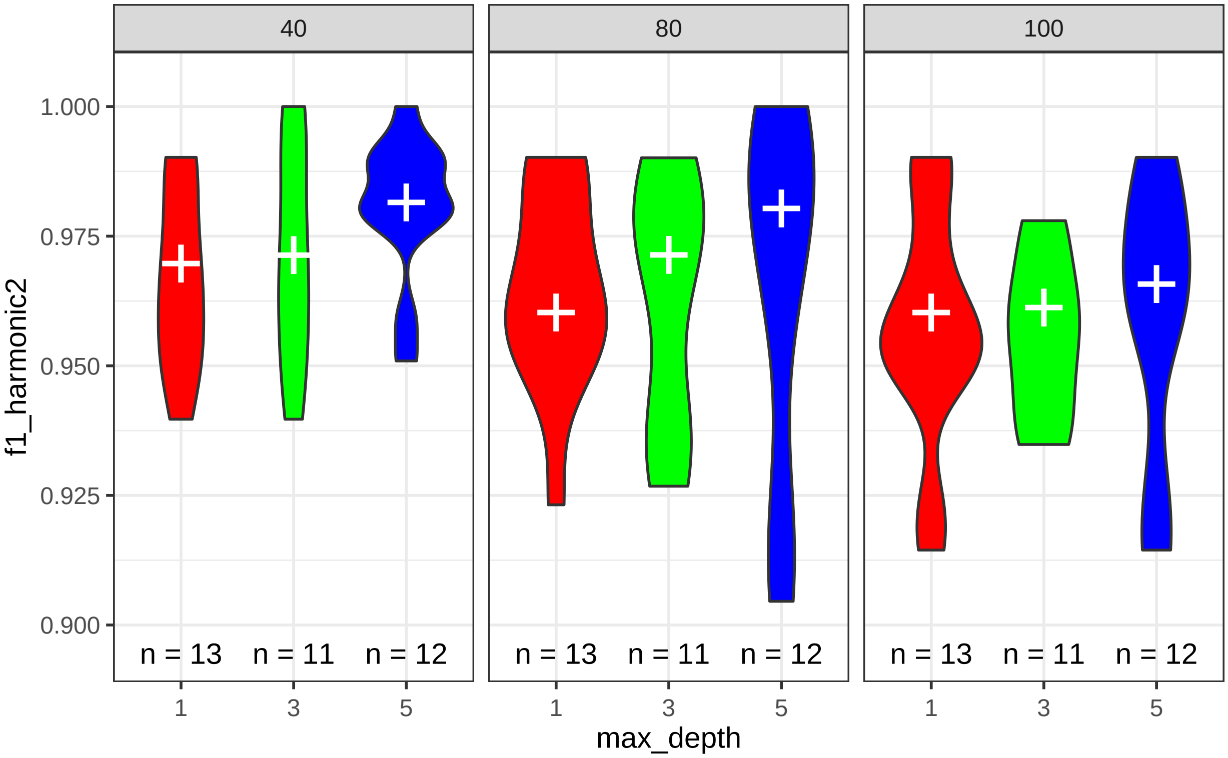

Top 20 features at 100 feature set
Gain (averaged across seeds)
Frequency (averaged across seeds)
1.2
0.9
0.6
0.3
0.0
0.00
0.25
0.50
0.75
1.00
GeneID.79083
GeneID.2625
GeneID.196993
GeneID.80129
GeneID.53335
GeneID.2099
GeneID.135295
GeneID.1058
GeneID.3169

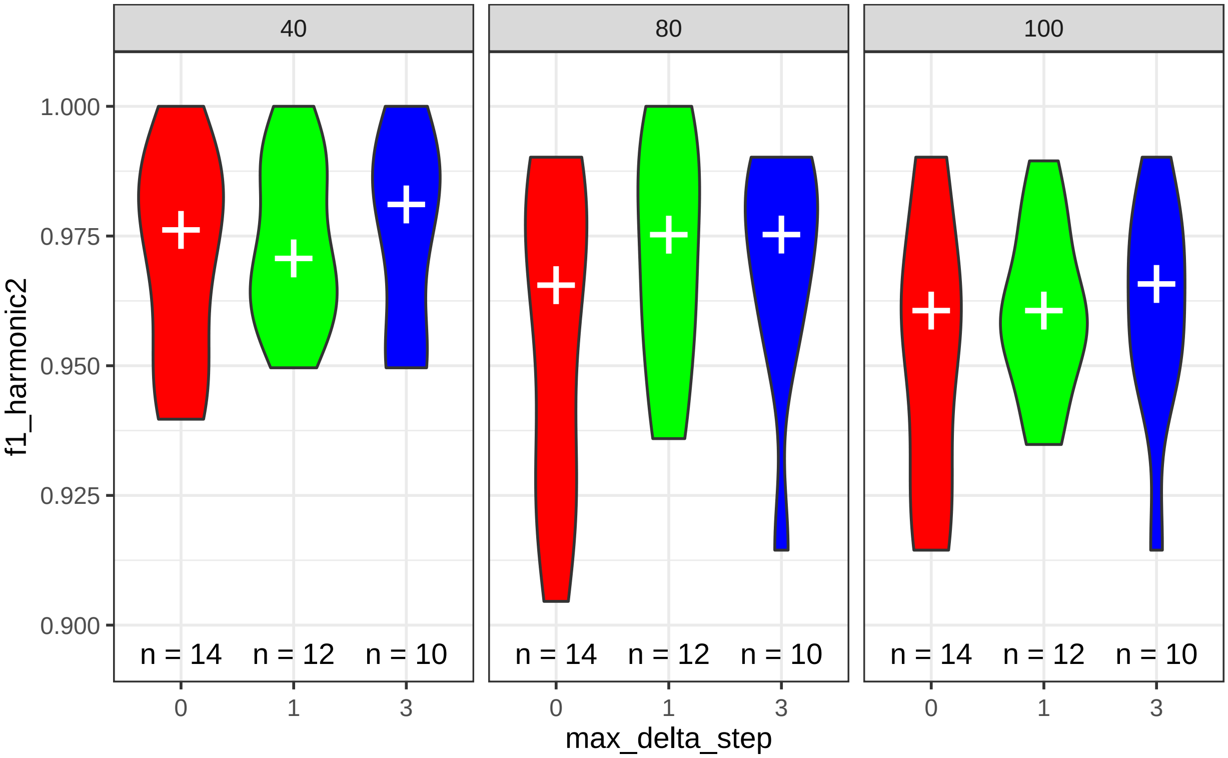


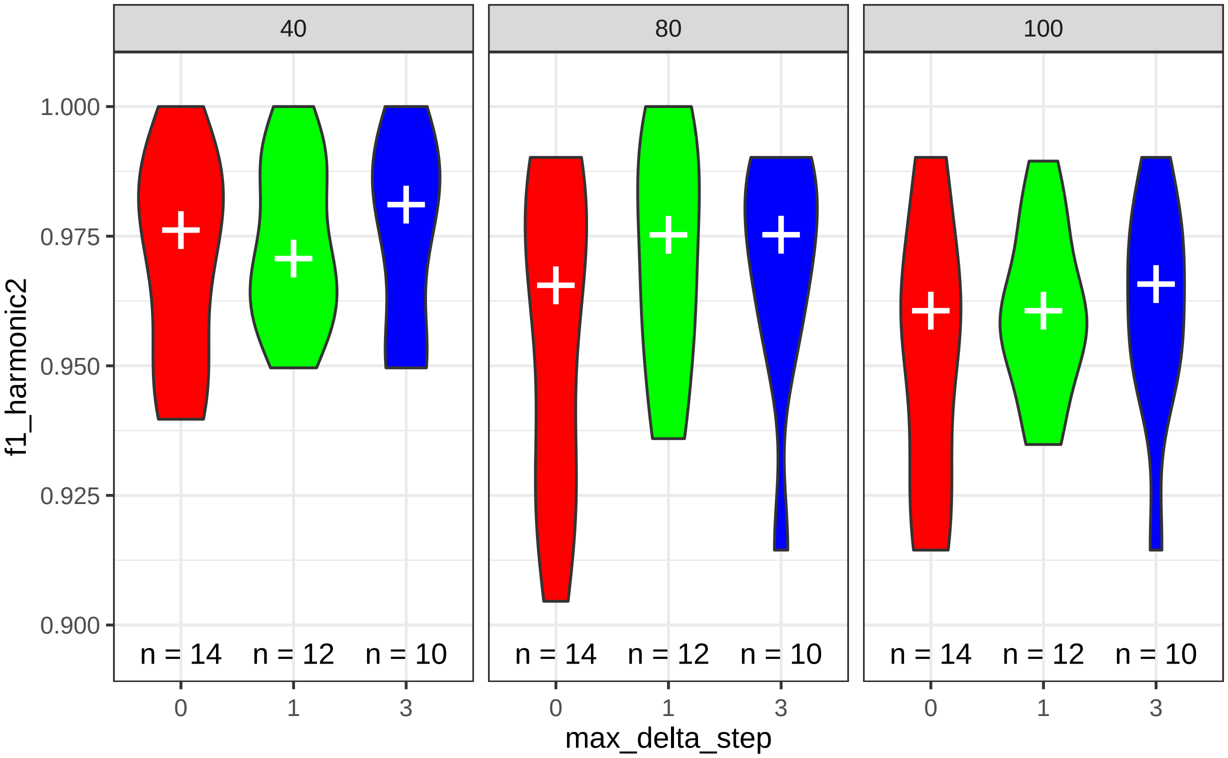
3. Hyper-parameters

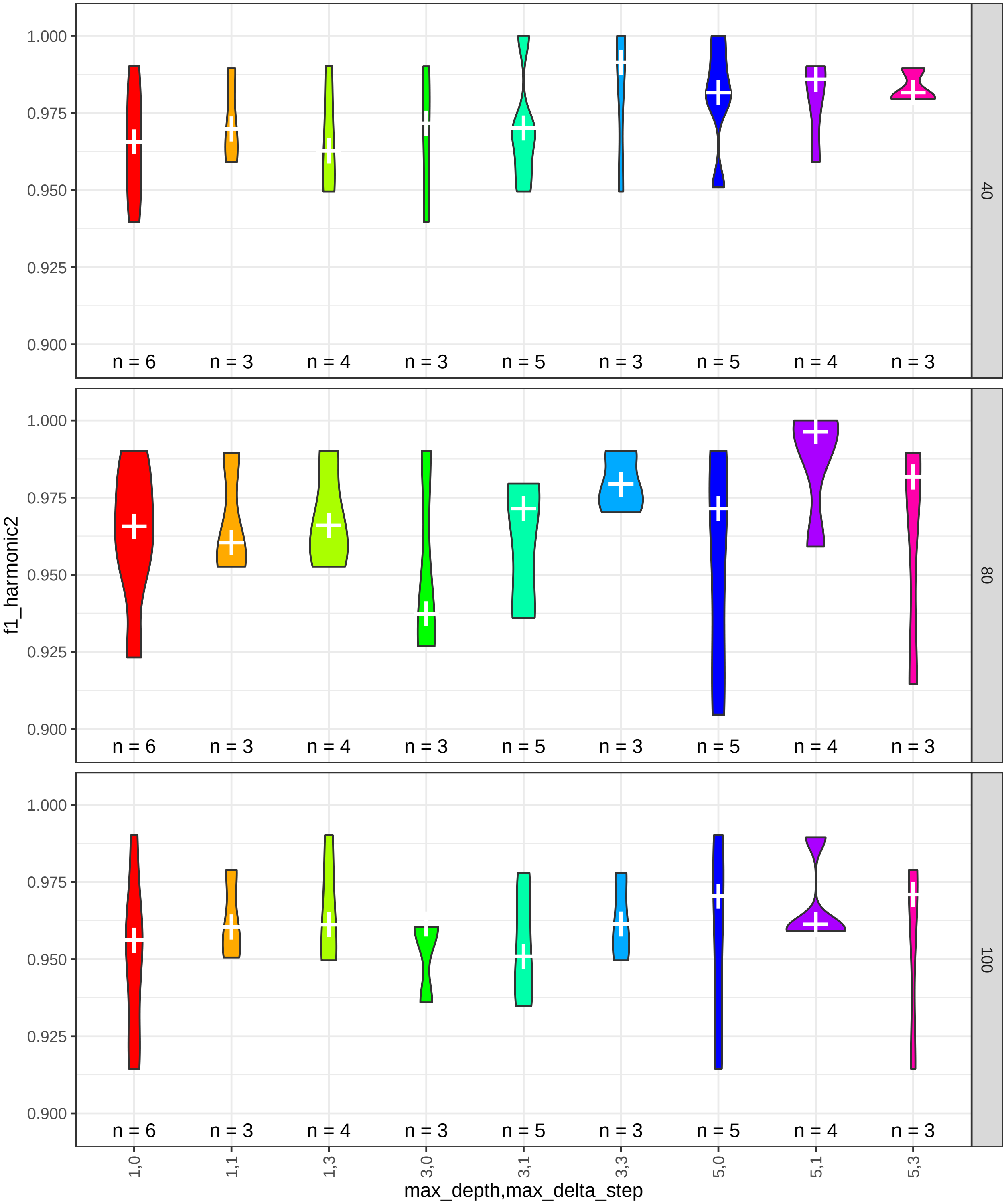
parameter optimization file (108 records) includes 4 seeds. Each seed generates 3 cv splits. Within each cv split, there is a 3 step RFE (at 40, 80, 100). So 108 / 4 / 3 / 3 = 3 parameter combinations tried in each cv split.

all grid search results



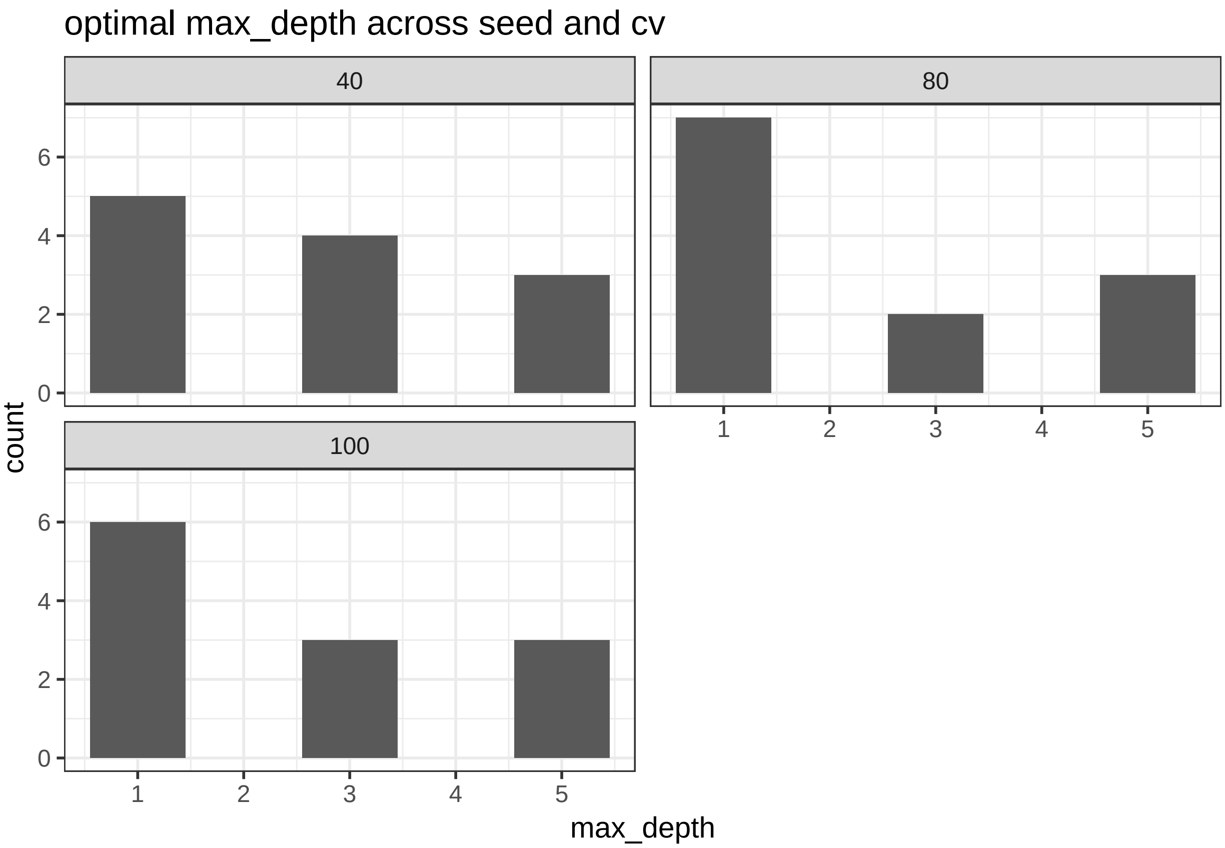


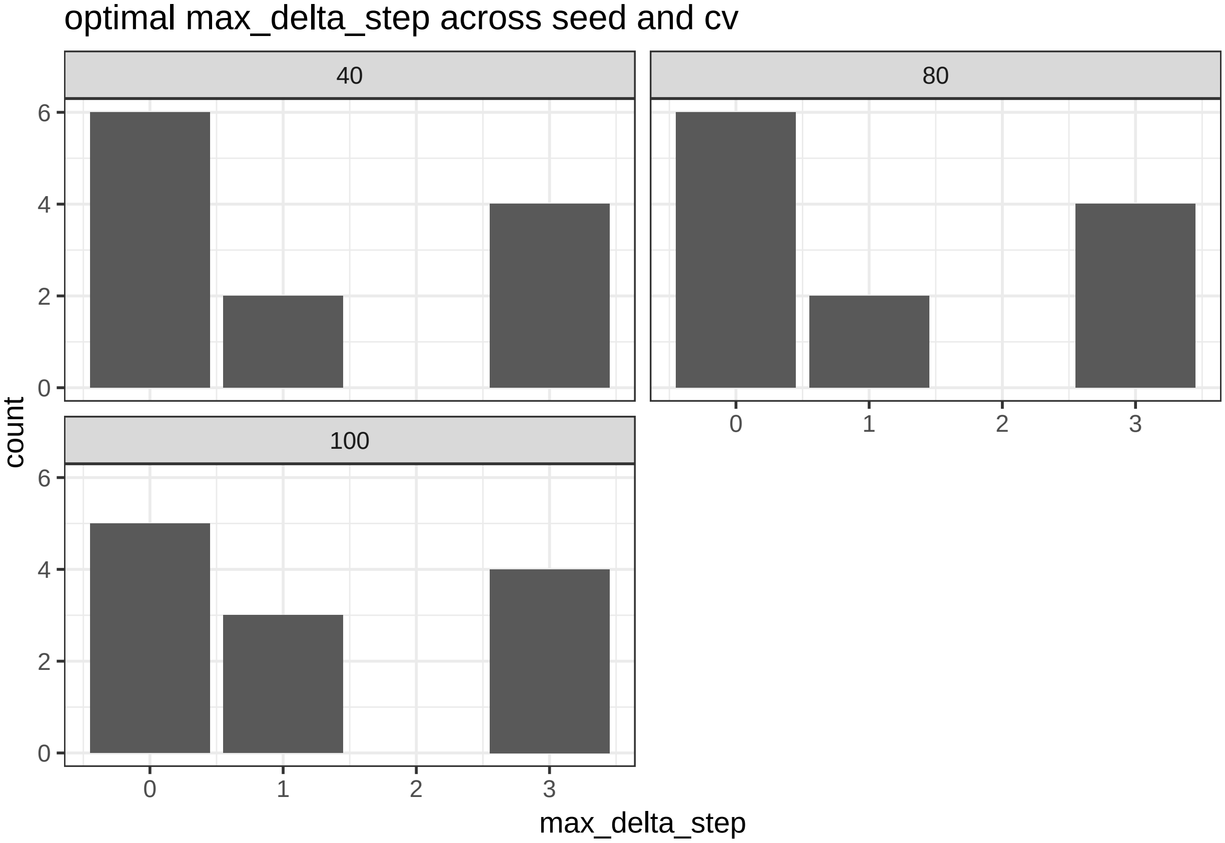

40
80
100
f1_harmonic2
1.000
0.975
0.950
0.925
0.900
n = 14
n = 12
n = 10
0
1
3
max_delta_step

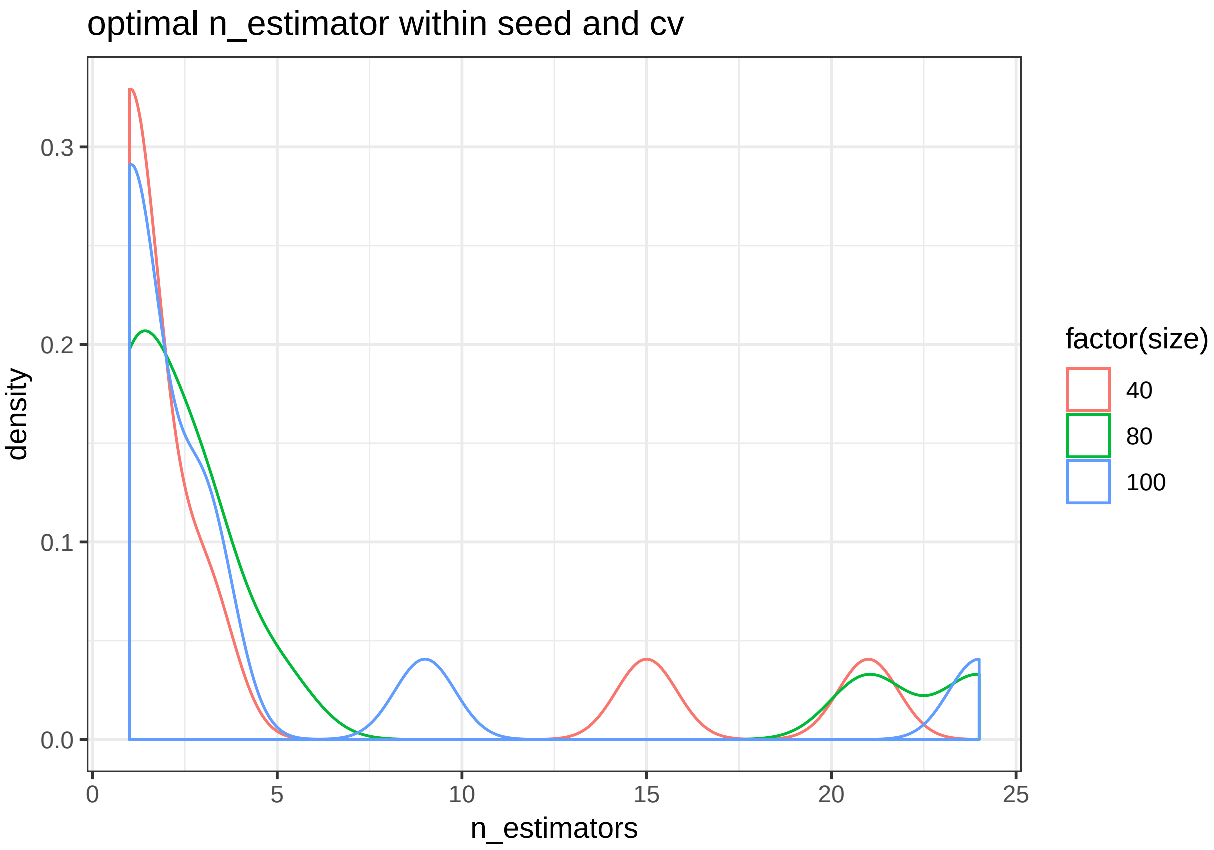


over best parameter combo per cv

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


optimal max_depth across seed and cv
40
80
100
count
0
2
4
6
1
2
3
4
5
max_depth


optimal max_delta_step across seed and cv
40
80
100
count
0
2
4
6
0
1
2
3
max_delta_step



more about the best parameter combination selection

```
select_ft_step <- 100

df1 <- subset(grid_best, size==select_ft_step & max_depth==1 &  max_delta_step == 0 )
print( paste('summary of n estimator at',select_ft_step, 'feature step'))
```

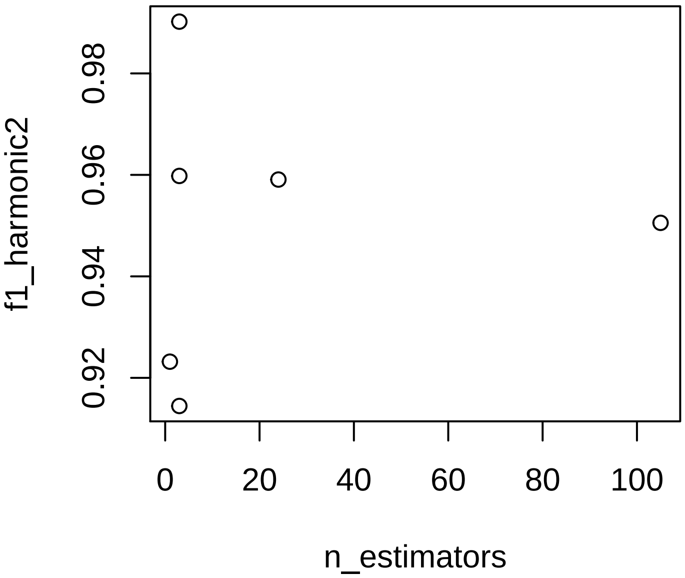
```
## [1] "summary of n estimator at 100 feature step"
```

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

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   9.333  13.500  24.000
```

```
df2 <- subset(df.grid, size==select_ft_step & max_depth==1 &  max_delta_step == 0 )

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


f1_harmonic2
0.92
0.94
0.96
0.98
0
20
40
60
80
100
n_estimators