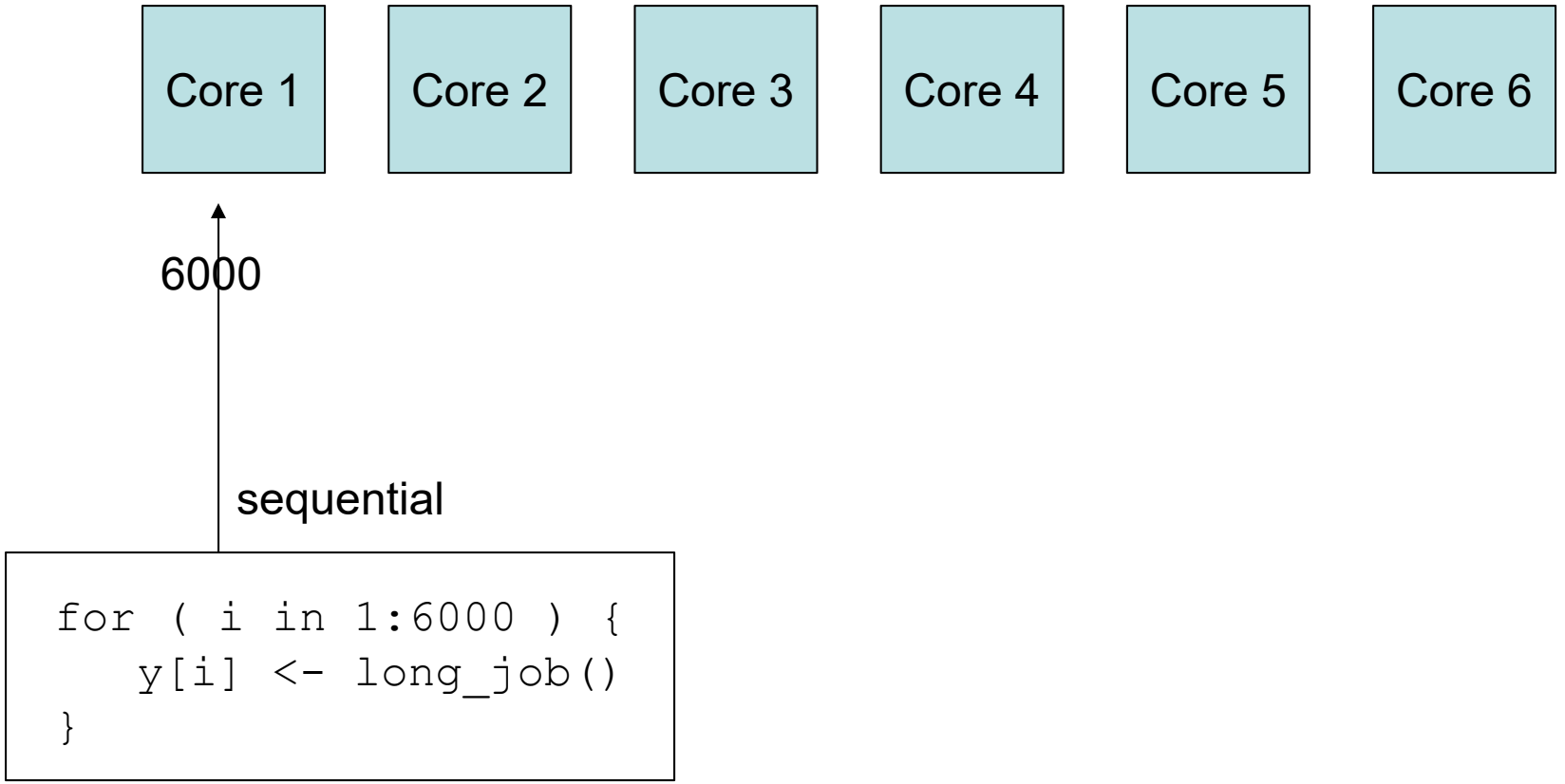
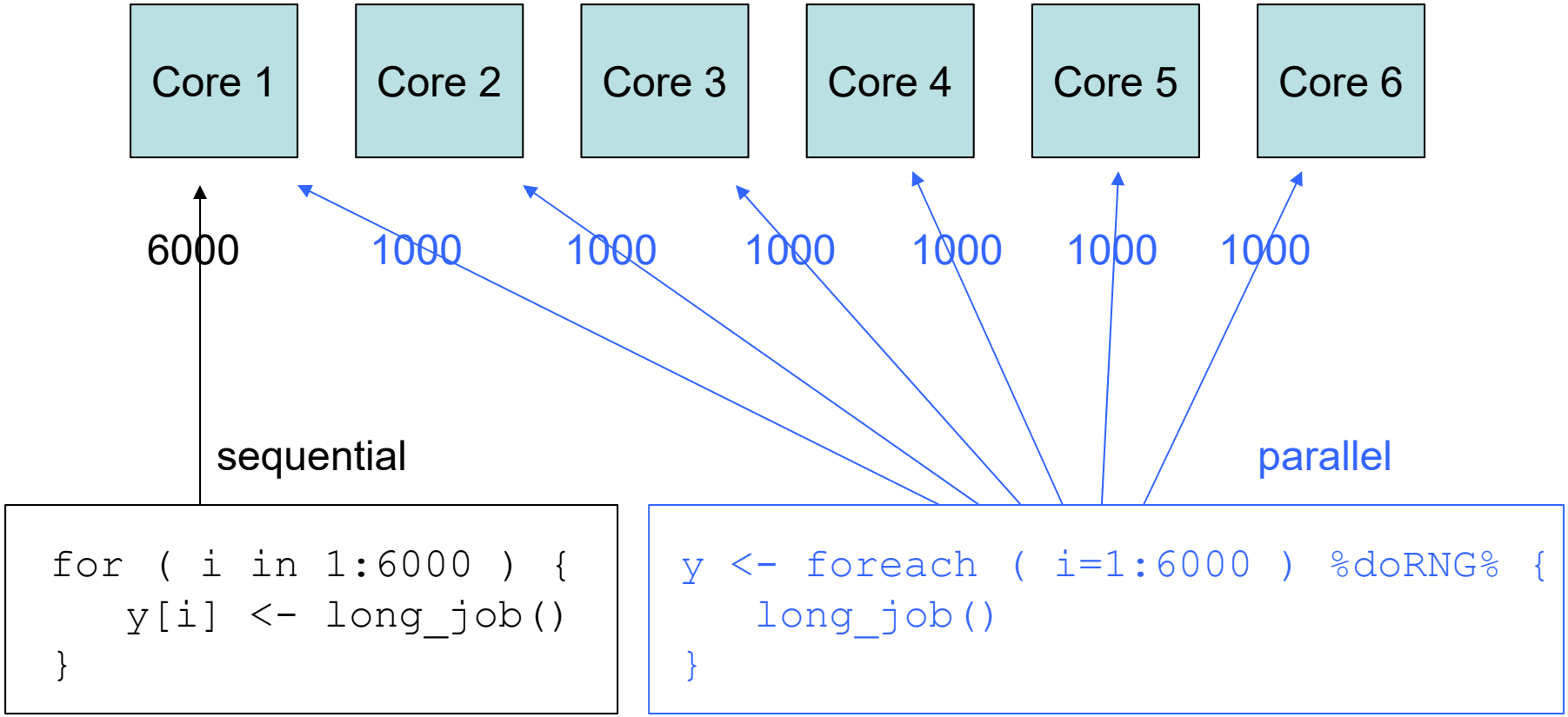
Today

- Bagging
  - see slides from last Thursday
  - code: ants_bag.R, ants_bag.py
  - inference algorithm (predictive performance)
  - tuning
- Parallel processing

Parallel processing



Parallel processing



Setup

```
library(doFuture)
library(doRNG)
registerDoFuture()
```



```
library(future)
library(foreach)
```

```
availableCores()
plan(multisession, workers=8)
```

```
# Handy timing function:
system.time( some_function() )

# Saving/loading long jobs
save(myresult1, myresult2, file="/saved/myresult.Rdata")
load("/saved/myresult.Rdata")
```

Inference algorithm

- 5-fold CV, 500 splits
- Ants: mean prediction error (MSE)

  12.93 +/- 0.07

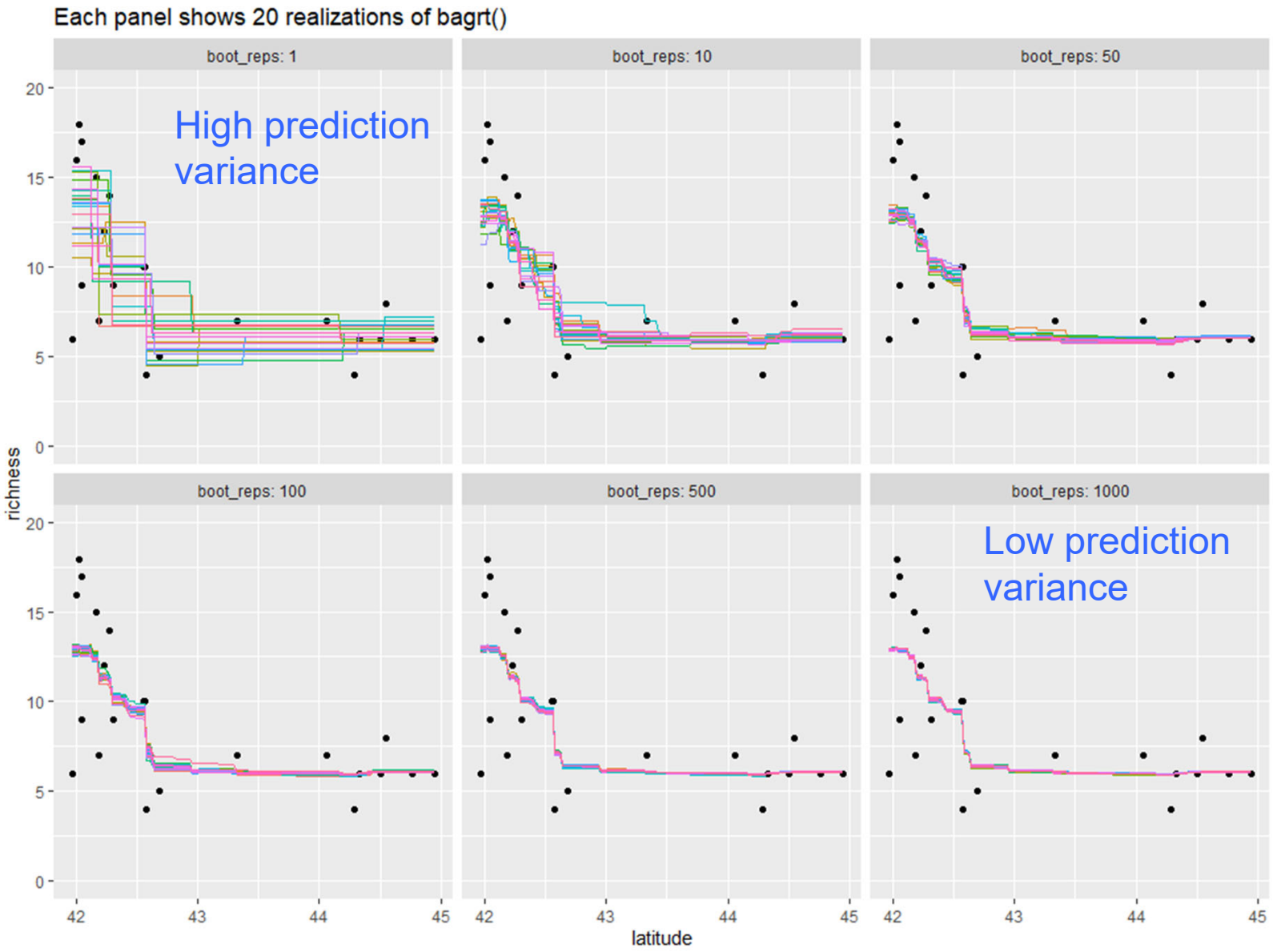
Model	LOOCV	5-fold CV
Polynomial 2	12.88	13.51
Single reg tree	12.68	13.15
KNN 7	12.63	13.03
KNN 6	12.95	13.01
Bagged reg tree	13.23*	12.93
Smoothing spline 3	12.52	12.77

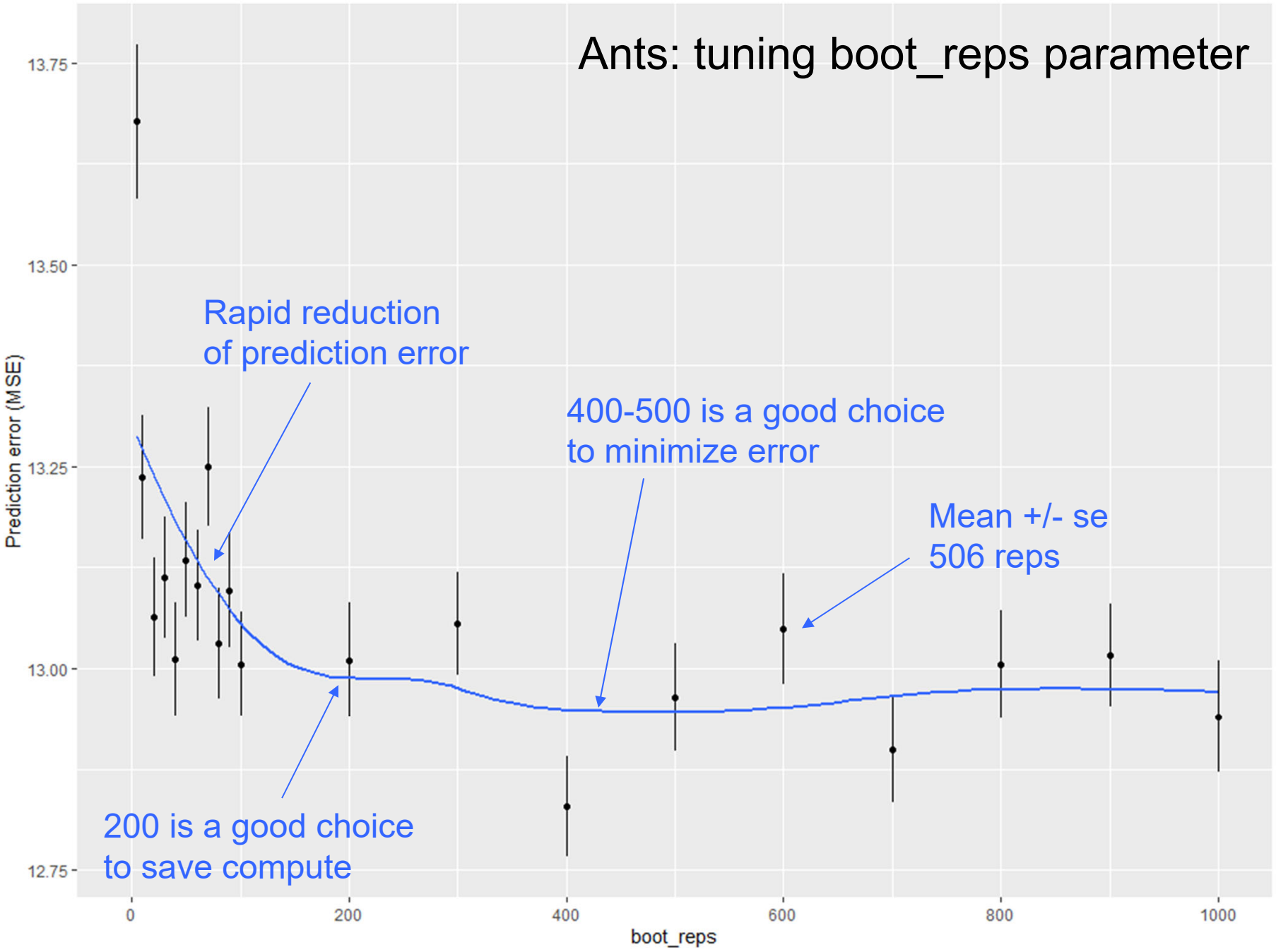
2nd best on 5-fold CV:
generally good predictive function

Worst on LOOCV:
perhaps an influential data point

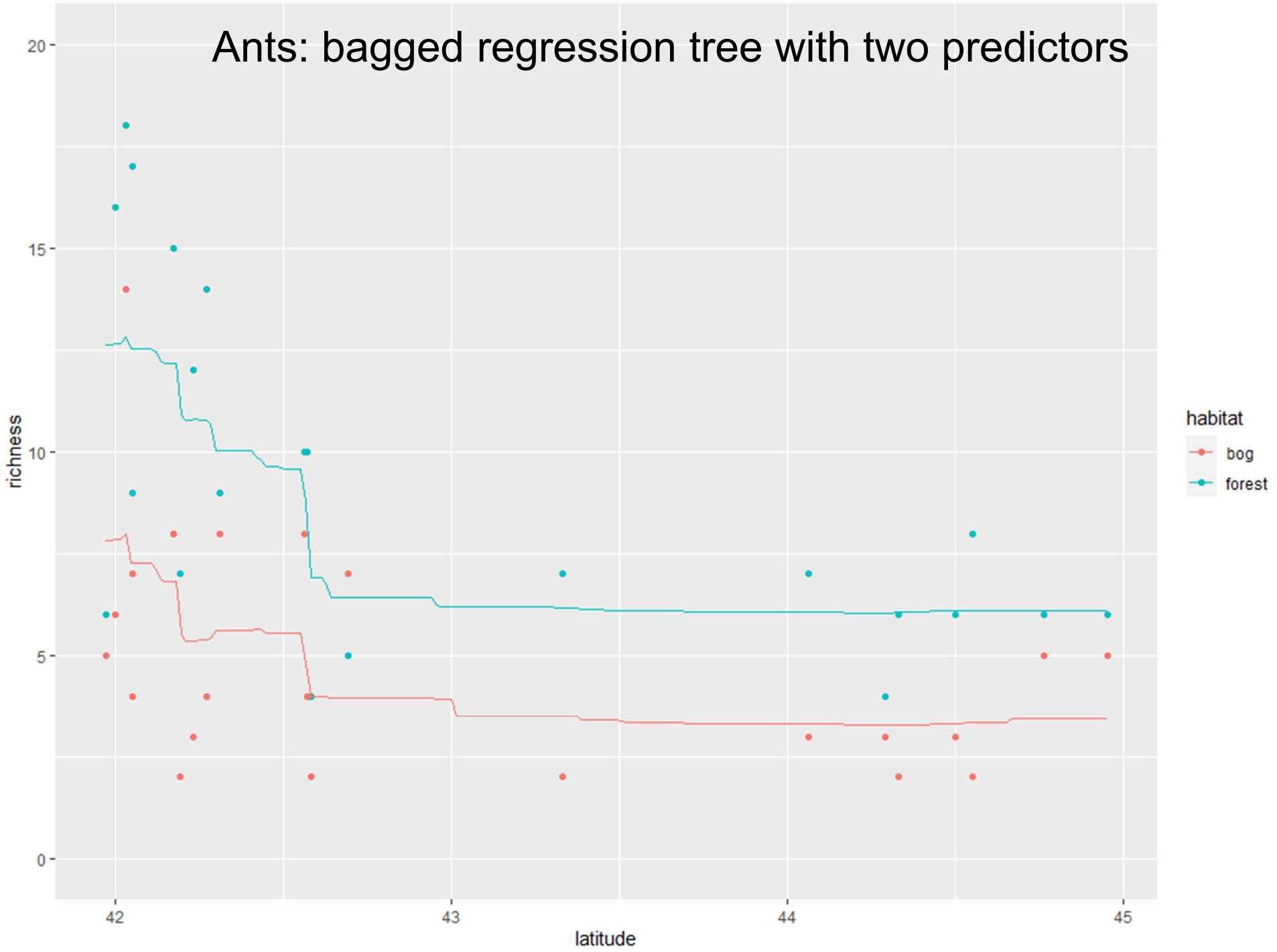
Prediction is always tenuous
with a small dataset

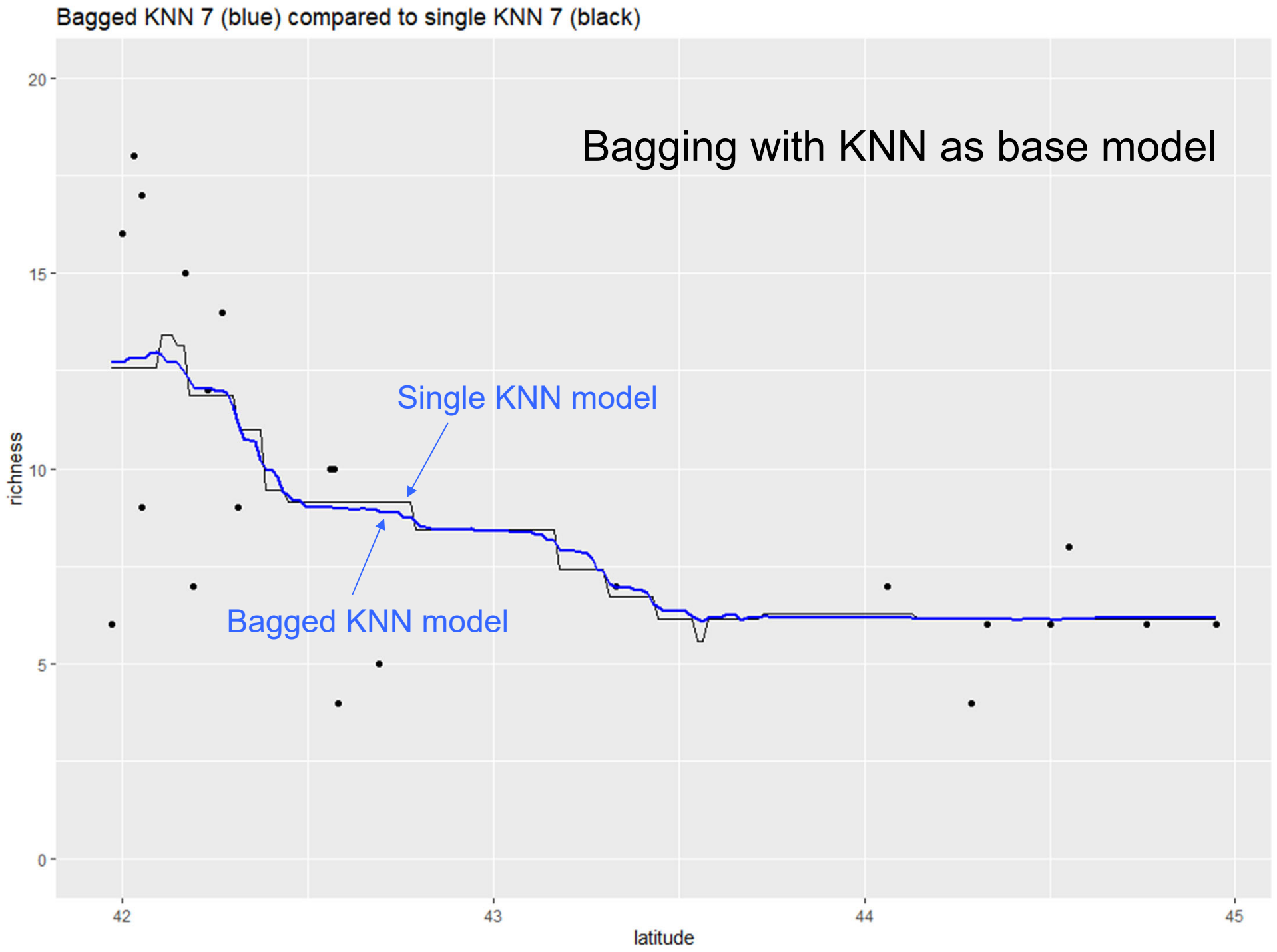
Bagging reduces prediction variance




Ants: tuning boot_reps parameter
Rapid reduction
of prediction error
400-500 is a good choice
to minimize error
Mean +/- se
506 reps
200 is a good choice
to save compute
Prediction error (MSE)
13.75
13.50
13.25
13.00
12.75
0
200
400
600
800
1000
boot_reps

Ants: bagged regression tree with two predictors




Bagged KNN 7 (blue) compared to single KNN 7 (black)
Bagging with KNN as base model
Single KNN model
Bagged KNN model
richness
latitude
20
15
10
5
0
42
43
44
45