

Validation set vs training set grad training - Approach two CIFAR100

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
##      filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      intersect, setdiff, setequal, union
```

```
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
library("ggpubr")
```

```
## Warning: package 'ggpubr' was built under R version 4.0.5
```

This experiment focuses on the question, whether training gradient based combining methods on the same set of data as the neural networks were trained on has any adverse effects to the performance of the ensemble as opposed to training on a different set, not presented to the networks during the training. Experiment was performed with two slightly different approaches.

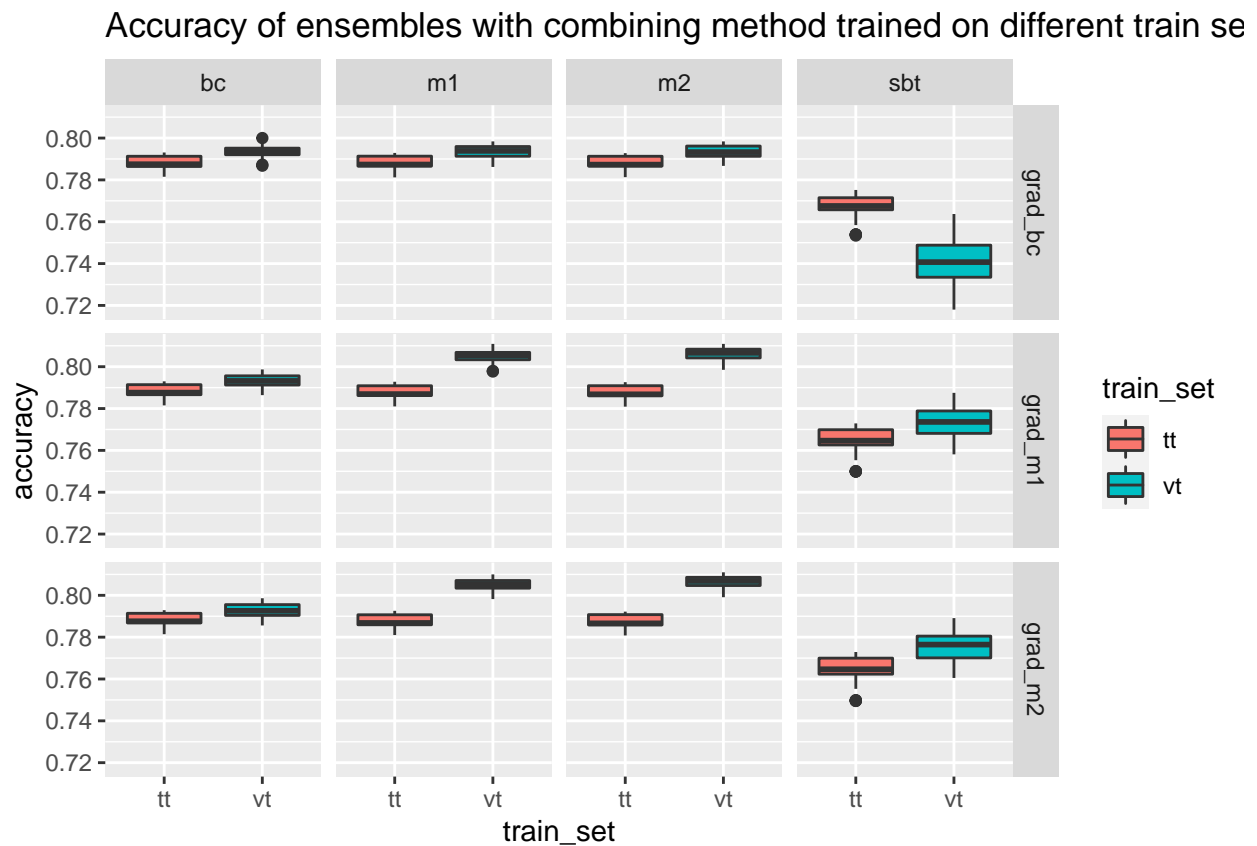
Approach two Experiment code is in the file `half_train_ensembling_experiment.py`. Experiment on both CIFAR10 and CIFAR100 datasets. This experiment differs from the previous one in the neural networks training part. In this case, the networks were trained on half of the original training set. The remainder of the training set was extracted as a validation set. This enabled us to train several ensembles on both the training set and the validation set in each replication. Experiment on CIFAR10 was performed in 1 replication and experiment on CIFAR100 in 10 replications. This is due to 10 times more classes in CIFAR100 and thus a need for 10 times larger training set in order to maintain constant 50 samples for class in combining method models training.

Approach one is described in the file `visualization_base_experiment`.

CIFAR100

```
net_results <- read.csv("../data/data_train_val_half_c100/net_metrics_grad.csv")
ens_results <- read.csv("../data/data_train_val_half_c100/ensemble_metrics_grad.csv")
```

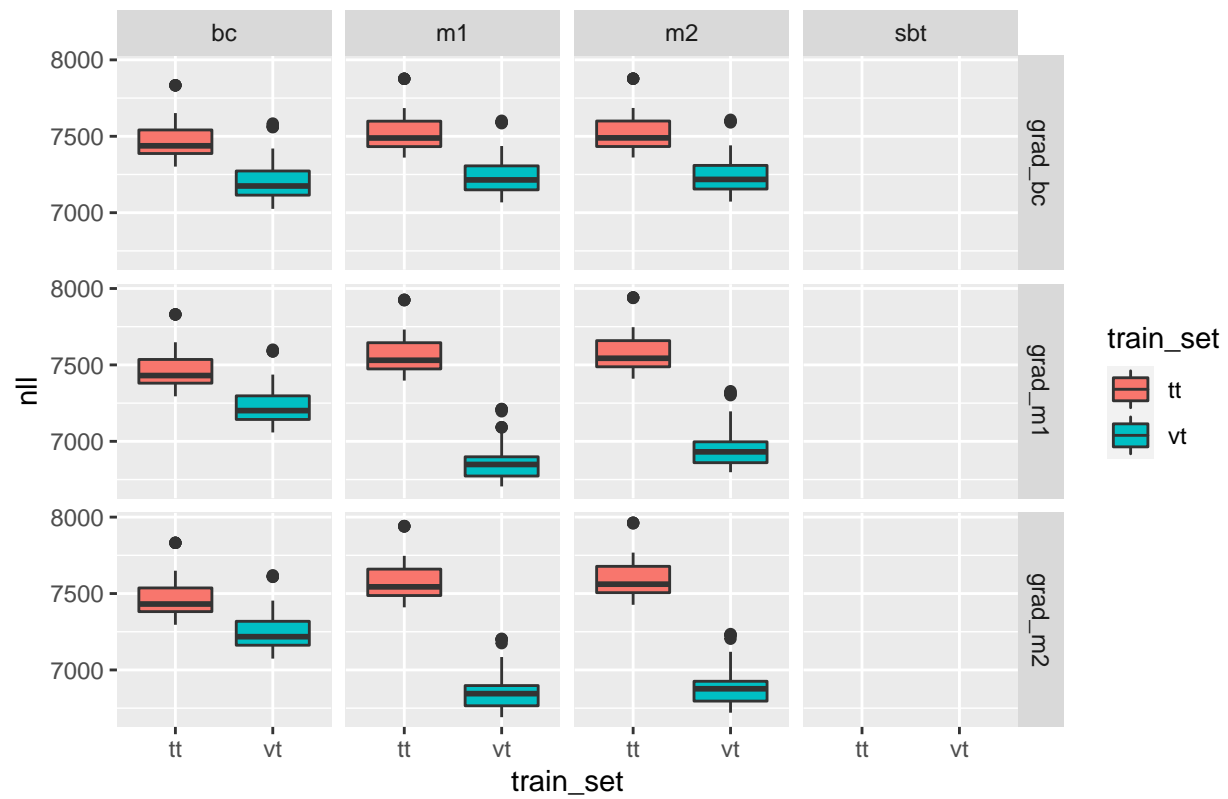
```
box_acc <- ggplot() + geom_boxplot(data=ens_results, mapping = aes(x=train_set, y=accuracy, fill=train_set),
  facet_grid(rows=vars(combining_method), cols=vars(coupling_method)) +
  ggtitle("Accuracy of ensembles with combining method trained on different train sets")
box_acc
```



```
box_nll <- ggplot() + geom_boxplot(data=ens_results, mapping = aes(x=train_set, y=nll, fill=train_set))
  facet_grid(rows=vars(combining_method), cols=vars(coupling_method)) +
  ggtitle("NLL of ensembles with combining method trained on different train sets")
box_nll
```

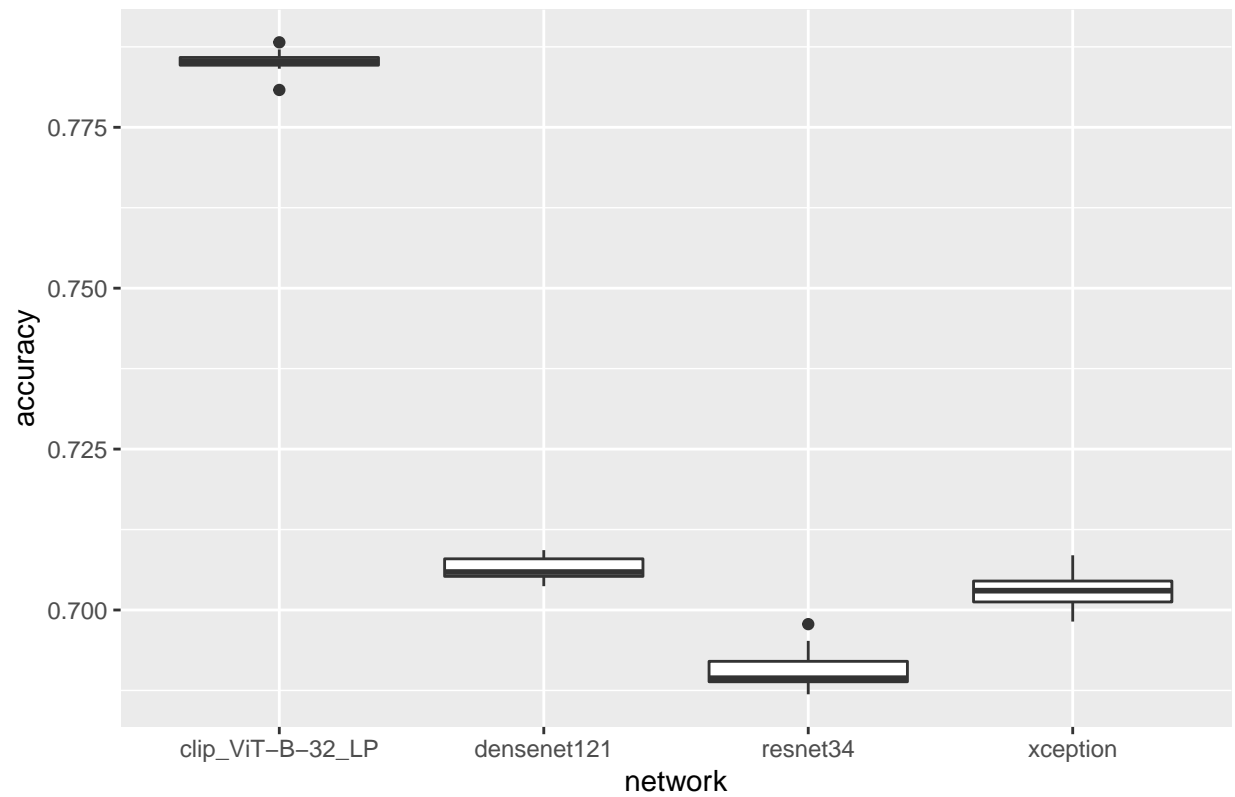
Warning: Removed 300 rows containing non-finite values (stat_boxplot).

NLL of ensembles with combining method trained on different train sets



```
box_net_acc <- ggplot() + geom_boxplot(data=net_results, mapping=aes(x=network, y=accuracy)) +
  ggtitle("Accuracy of networks")
box_net_acc
```

Accuracy of networks



```
box_net_nll <- ggplot() + geom_boxplot(data=net_results, mapping=aes(x=network, y=nll)) +
  ggtitle("NNL of networks")
box_net_nll
```

NNL of networks

