EWGT-2021: Potential of vision-enhanced floating car data for urban traffic estimation

Dmitry Pavlyuk

December 5, 2020

This markdown reproduces the research "Potential of vision-enhanced floating car data for urban traffic estimation, submitted to the 24rd EURO Working Group on Transportation Meeting, EWGT 2021.

Load necessary libraries

```
require(needs)
```

```
## Loading required package: needs
```

```
needs(igraph)
needs(matrixcalc)
needs(fsMTS)
needs(plot.matrix)
needs(expm)
needs(dplyr)
needs(ggplot2)
needs(tidyr)
needs(scales)
source("functions.R")
```

Load necessary libraries

```
data.folder <- file.path(getwd(),"data")
output.folder <- file.path(getwd(),"output")
```

Prepare PeMS data

```
pems.rds <- file.path(data.folder, "PeMS", "prepared.rds")
if (!file.exists(pems.rds)) {
  meta <-
    read.csv(
      file.path(data.folder, "PeMS", "d07_text_meta_2020_11_16.txt"),
      header = T,
      sep = "\t"
    )
  stations <-
    c(
      717046,
      717045,
      717263,
      717264,
```

```
    716943,
    716942,
    716331,
    717445,
    717047,
    716028,
    716946,
    718085,
    718173,
    716939
  )
observable <-
  list(
    '717046' = '717045',
    '717045' = '717046',
    '717263' = '717264',
    '717264' = '717263',
    '716943' = '716942',
    '716942' = '716943',
    '716331' = '717445',
    '717445' = '716331',
    '717047' = '716028',
    '716028' = '717047',
    '716946' = '718085',
    '718085' = '716946',
    '718173' = '716939',
    '716939' = '718173'
  )
data.raw <-
  read.csv(file.path(
    data.folder,
    "PeMS",
    "d07_text_station_5min_2020_11_29.txt"
  ),
  header = F)
data.tb <- as_tibble(data.raw)
data.tb <-
  data.tb %>% mutate(
    datetime = as.POSIXct(V1, format = "%m/%d/%Y %H:%M:%S"),
    station = V2,
    volume = V10,
    occupancy = V11,
    speed = V12
  ) %>%
  select(datetime, station, volume, occupancy, speed)
data.tbf <-
  data.tb %>% filter(station %in% stations) %>%
  mutate(speed = ifelse(is.na(speed), 65, speed)) %>%

  mutate(volume = ifelse(is.na(volume), 0, volume))


vols <-
```

```
    data.tbf %>% select(datetime, station, volume) %>%
    pivot_wider(names_from = station, values_from = volume)
  rels <- cor(vols %>% select(-datetime))
  md <- rowSums(abs(rels))
  Lw <- -abs(rels) + diag(ncol(rels)) + diag(md)
  data.prepared <-
    list(
      traffic.data = data.tbf %>% mutate(station = as.factor(station)) %>%
        select(datetime, station, speed, volume),
      Lw = Lw,
      observable = observable
    )
  saveRDS(data.prepared, file = pems.rds)
} else{
  warning("Prepared data exists: loading")
  data.prepared <- readRDS(file = pems.rds)
}
```

```
## Warning: Prepared data exists: loading
```

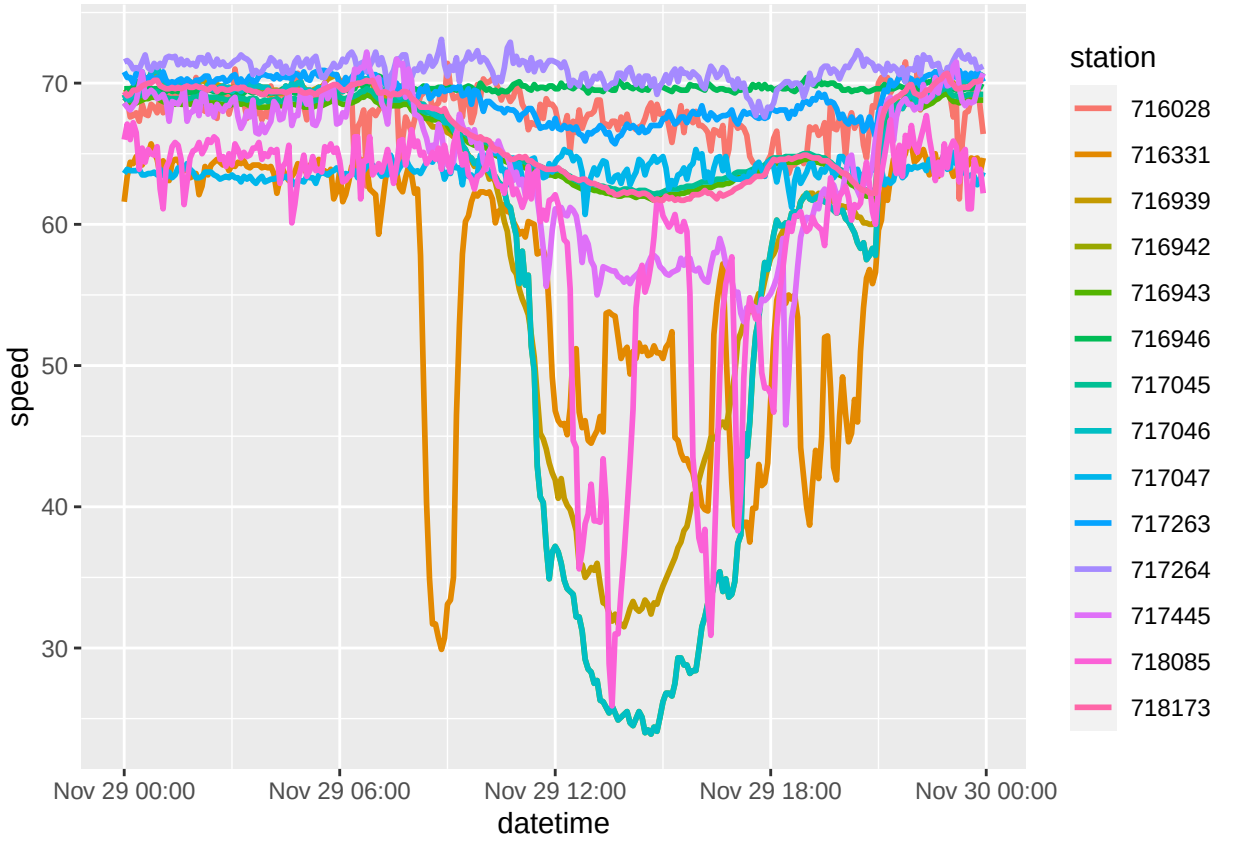
Descriptive analysis

```
data.speed <-
  data.prepared$traffic.data %>% select(datetime, station, speed) %>%
  pivot_wider(names_from = station, values_from = speed) %>% select(-datetime)
data.volume <-

  data.prepared$traffic.data %>% select(datetime, station, volume) %>%
  pivot_wider(names_from = station, values_from = volume) %>% select(-datetime)
N <- ncol(data.speed)
T <- nrow(data.speed)
```

Data dimensions: 288, 14

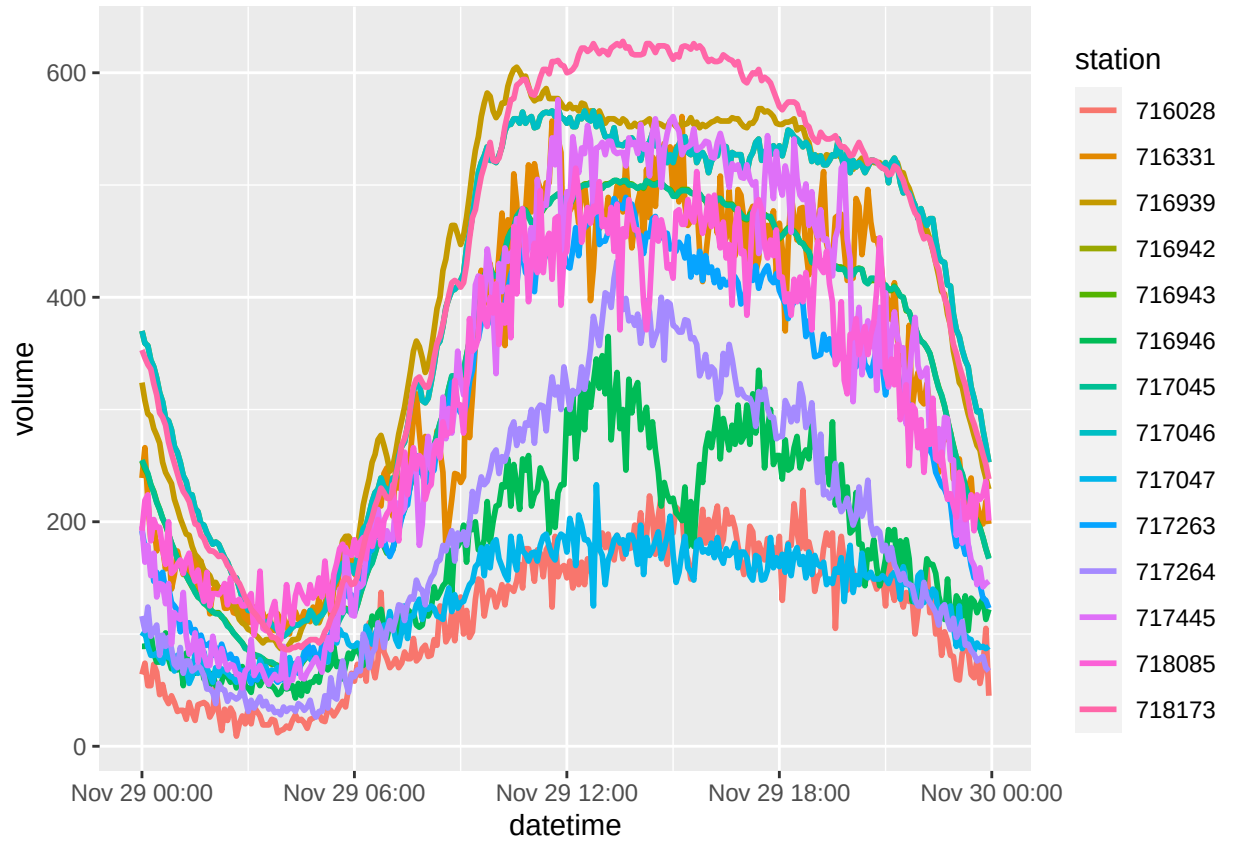
Speed plots

```
data.prepared$traffic.data %>% ggplot(aes(
  x = datetime,
  y = speed,
  col = station,
  group = station
)) + geom_line(size = 1)
```



Volume plots

```
data.prepared$traffic.data %>% ggplot(aes(
  x = datetime,
  y = volume,
  col = station,
  group = station
)) + geom_line(size = 1)
```



Speed variance

```
data.prepared$traffic.data %>% group_by(station) %>% summarise(sd(speed))
```

```
## # A tibble: 14 x 2
##    station `sd(speed)`
##    <fct>         <dbl>
##  1 716028        1.67
##  2 716331        9.39
##  3 716939       12.8
##  4 716942       16.2
##  5 716943        2.68
##  6 716946        0.260
##  7 717045        2.68
##  8 717046       16.2
##  9 717047        0.640
## 10 717263        1.47
## 11 717264        0.853
## 12 717445        5.49
## 13 718085        8.68
## 14 718173        3.14
```

Traffic estimation

```
sp <- 9e-4
omega <- randomOmega(N, T, sp, t(data.volume))
```

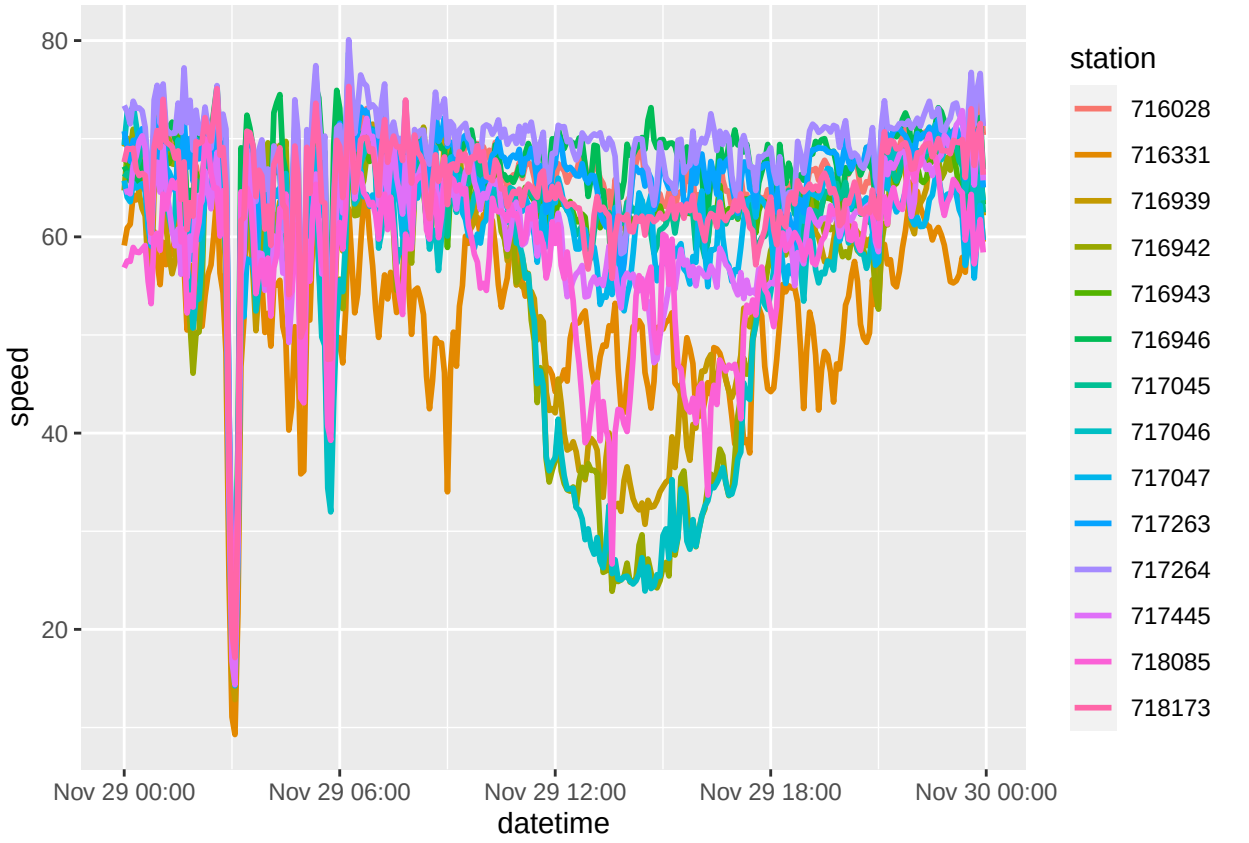
```r
sum(omega > 0) / (N * T)
```

```
## [1] 0.2264385
```

```r
res <-
  TGMCS(t(data.speed),
        data.prepared$Lw,
        omega,
        returnQhat = T,
        lambda3 = 1)
```

```
## [1] "converged =  TRUE"
```

```r
hat.tbf <- as_tibble(t(res$Qhat))
hat.tbf$datetime <-
  unique(data.prepared$traffic.data %>% pull(datetime))
hat.tbf %>% pivot_longer(-one_of("datetime"),
                         names_to = "station",
                         values_to = "speed") %>%
  ggplot(aes(
    x = datetime,
    y = speed,
    col = station,
    group = station
  )) + geom_line(size = 1)
```




```r
res$observedMAE
```

```
## [1] 0.1202376
```

```
res$unobservedMAE
```

```
## [1] 4.779484
```

```
res$observedMAPE
```

```
## [1] 0.002164136
```

```
res$unobservedMAPE
```

```
## [1] 0.07498343
```

Experiments

```
results.rds <- file.path(output.folder, "results.rds")
sp.list <-
  c(4e-5, 1e-4, 2e-4, 3e-4, 4e-4, 5e-4, 7e-4, 9e-4, 13e-4, 21e-4, 27e-4)
sp.list <- c(2e-4, 3e-4, 4e-4, 5e-4, 7e-4, 9e-4, 13e-4)
for (sp in sp.list) {
  omega <- randomOmega(N, T, sp, t(data.volume))
  print(paste(sp, "=", round(sum(omega > 0) / (N * T), 2)))
}
```

```
## [1] "2e-04 = 0.06"
## [1] "3e-04 = 0.09"
## [1] "4e-04 = 0.13"
## [1] "5e-04 = 0.14"
## [1] "7e-04 = 0.19"
## [1] "9e-04 = 0.24"
## [1] "0.0013 = 0.29"
```

```
if (!file.exists(results.rds)) {
  est <- list()
  r <- 15
  mu = 0.0001
  lambda1 = 0.01
  lambda2 = 0.05
  lambda3 = 1
  tol = 1e-6
  maxIter = 1e6
  for (rep in 1:50) {
    for (sp in sp.list) {
      print(paste(rep, sp))
      omega <- randomOmega(N, T, sp, t(data.volume))
      omegax2 <- omega + randomOmega(N, T, sp, t(data.volume))
      omegax2[omegax2 > 1] <- 1
      omegaExt <- enhanceOmega(omega, data.prepared$observable)
      est[[length(est) + 1]] <-
        c(
          TGMCS(
            t(data.speed),
            data.prepared$Lw,
            omega,
            accMask = omega,
```

```
      maxIter = maxIter,
      tol = tol,
      r = r,
      mu = mu,
      lambda1 = lambda1,
      lambda2 = lambda2,
      lambda3 = lambda3
    ),
    sparsity = sp,
    obslinks = sum(omega > 0),
    coverage = sum(omega > 0) / (N * T),
    name = "omega"
  )
est[[length(est) + 1]] <-
  c(
    TGMCS(
      t(data.speed),
      data.prepared$Lw,
      omegax2,
      accMask = omegax2,
      maxIter = maxIter,
      tol = tol,
      r = r,
      mu = mu,
      lambda1 = lambda1,
      lambda2 = lambda2,
      lambda3 = lambda3
    ),
    sparsity = sp,
    obslinks = sum(omegax2 > 0),
    coverage = sum(omegax2 > 0) / (N * T),
    name = "omegax2"
  )
est[[length(est) + 1]] <-
  c(
    TGMCS(
      t(data.speed),
      data.prepared$Lw,
      omegaExt,
      accMask = omegaExt,
      maxIter = maxIter,
      tol = tol,
      r = r,
      mu = mu,
      lambda1 = lambda1,
      lambda2 = lambda2,
      lambda3 = lambda3
    ),
    sparsity = sp,
    obslinks = sum(omegaExt > 0),
    coverage = sum(omegaExt > 0) / (N * T),
    name = "omegaExt"
  )
```

```
      print(tail(bind_rows(est)))
    }
    saveRDS(est, file = results.rds)
  }
} else{
  est <- readRDS(results.rds)
}
```

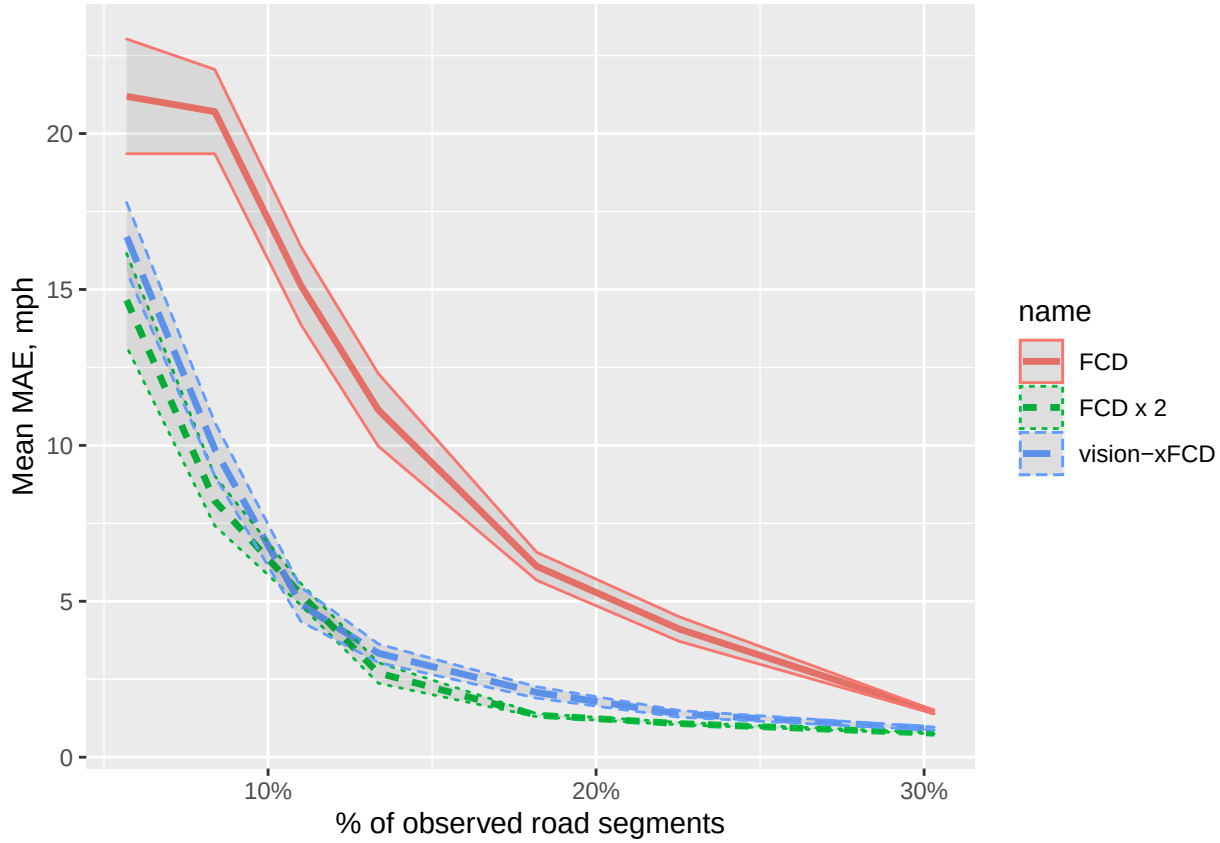
Experimental results

```
est.df <- bind_rows(est)
est.df
```

```
## # A tibble: 954 x 9
##    observedMAE unobservedMAE observedMAPE unobservedMAPE converged sparsity
##          <dbl>         <dbl>        <dbl>          <dbl> <lgl>        <dbl>
##  1       0.145          32.0      0.00240          0.493 TRUE      0.00004
##  2       0.152          19.8      0.00263          0.313 TRUE      0.00004
##  3       0.166          28.8      0.00305          0.442 TRUE      0.00004
##  4       0.153          18.7      0.00293          0.291 TRUE      0.0001
##  5       0.162          26.1      0.00303          0.397 TRUE      0.0001
##  6       0.157          17.1      0.00291          0.260 TRUE      0.0001
##  7       0.174          23.7      0.00329          0.362 TRUE      0.0002
##  8       0.128           8.04     0.00246          0.128 TRUE      0.0002
##  9       0.141          12.0      0.00259          0.184 TRUE      0.0002
## 10       0.152          25.7      0.00274          0.393 TRUE      0.000300
## # ... with 944 more rows, and 3 more variables: obslinks <int>, coverage <dbl>,
## #   name <chr>
```

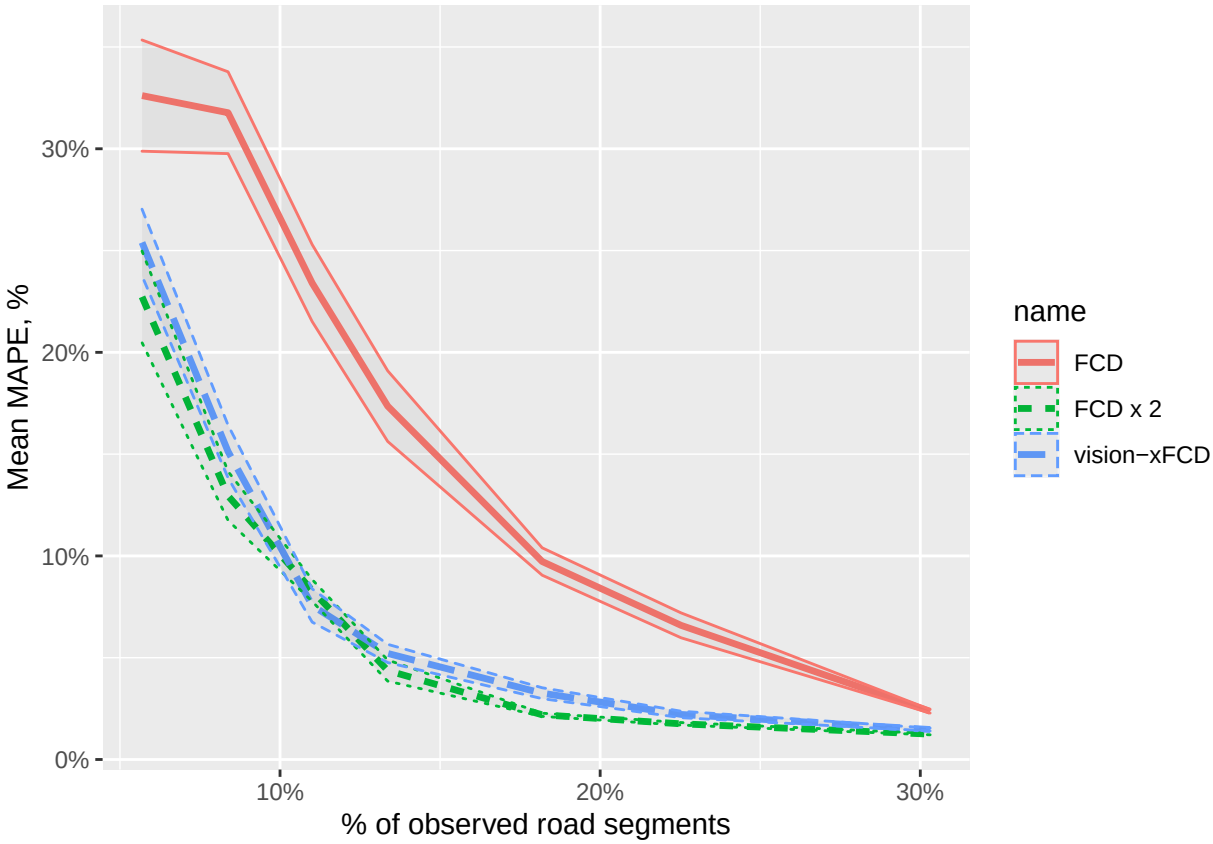
```
mCov.df <-
  est.df %>% filter(name == "omega") %>% group_by(sparsity) %>%
  summarise(meanCoverage = mean(coverage))
est.df <- est.df %>% left_join(mCov.df, by = c("sparsity"))

f <- 1.96
legend_names <-
  c('omega' = 'FCD',
    'omegax2' = 'FCD x 2',
    'omegaExt' = 'vision-xFCD')
est.df %>% filter(converged == TRUE, sparsity > 1e-4, sparsity <= 13e-4) %>%
  mutate(name = legend_names[name]) %>%
  group_by(name, sparsity, meanCoverage) %>%
  summarise(
    meanMAE = mean(unobservedMAE),
    sdMAE = sd(unobservedMAE),
    n = n(),
    minMAE = min(unobservedMAE),
    maxMAE = max(unobservedMAE),
    lb = max(meanMAE - f * sdMAE / sqrt(n), minMAE),
    ub = min(meanMAE + f * sdMAE / sqrt(n), maxMAE)
  ) %>%
  ggplot(aes(
    x = meanCoverage,
    y = meanMAE,
    col = name,
```

```
    group = name,
    linetype = name
  )) + geom_line(size = 1.2) +
  geom_ribbon(aes(
    ymin = lb,
    ymax = ub,
    col = name,
    group = name,
    linetype = name
  ),
  alpha = 0.1) +
  scale_x_continuous(
    labels = function(x)
      paste0(x * 100, "%")
  ) + labs(x = "% of observed road segments", y = "Mean MAE, mph")
```




```
est.df %>% filter(converged == TRUE, sparsity > 1e-4, sparsity <= 13e-4) %>%
  mutate(name = legend_names[name]) %>%
  group_by(name, sparsity, meanCoverage) %>%
  summarise(
    meanMAPE = mean(unobservedMAPE),
    sdMAPE = sd(unobservedMAPE),
    n = n(),
    minMAPE = min(unobservedMAPE),
    maxMAPE = max(unobservedMAPE),
    lb = max(meanMAPE - f * sdMAPE / sqrt(n), minMAPE),
```

```
    ub = min(meanMAPE + f * sdMAPE / sqrt(n), maxMAPE)
) %>%
ggplot(aes(
    x = meanCoverage,
    y = meanMAPE,
    col = name,
    group = name,
    linetype = name
)) + geom_line(size = 1.2) +
geom_ribbon(aes(
    ymin = lb,
    ymax = ub,
    col = name,
    group = name,
    linetype = name
),
alpha = 0.05) +
scale_y_continuous(
    labels = function(x)
        paste0(x * 100, "%")
) +
scale_x_continuous(
    labels = function(x)
        paste0(x * 100, "%")
) + labs(x = "% of observed road segments", y = "Mean MAPE, %")
```



```r
est.df %>% filter(converged == TRUE, sparsity > 1e-4, sparsity <= 13e-4) %>%
  mutate(name = legend_names[name]) %>%
  group_by(name, sparsity, meanCoverage) %>%
  summarise(meanMAPE = mean(unobservedMAPE) * 100,
            meanSp = mean(obslinks) / (N * T)) %>%
  mutate(coverage = paste0(round(meanCoverage * 100), "%")) %>%
  pivot_wider(
    id_cols = c(name),
    names_from = "coverage",
    values_from = "meanMAPE"
  )
```

```
## # A tibble: 3 x 8
## # Groups:   name [3]
##   name          `6%`  `8%` `11%` `13%` `18%` `23%` `30%`
##   <chr>        <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 FCD           32.6  31.8 23.4  17.4   9.73  6.59  2.37
## 2 FCD x 2       22.7  12.9  8.30  4.37  2.19  1.75  1.25
## 3 vision-xFCD   25.4  15.1  7.56  5.21  3.26  2.21  1.47
```