

Exploratory Plot

Ben Sherwood

2022-10-25

Getting data

```
allData <- NULL
for(n in c(100,500,1000)){
  allData <- rbind(allData, readRDS(paste0("NormalWild_n_",n,"_ycoef_-1_2022.RDS"))$results)
}
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.1.1
library(reshape)

## Warning: package 'reshape' was built under R version 4.1.3
allData$Method <- paste(allData$Method,allData$x1Type)
```

MSE Table

```
mseTable <- allData[,c(1,3,5,7:10)]
mseTable <- cbind(mseTable, apply(allData[,11:14],1,sum), allData[,15])
for(i in 4:9){
  mseTable[,i] <- round(mseTable[,i],4)
}
mseTable
```

##	Method	m	n	B1	B2	B3	B4	apply(allData[, 11:14], 1, sum)
## 1	Oracle	2	100	0.1689	0.0141	0.0226	0.0073	0.8341
## 2	Naive	2	100	0.0452	0.1495	0.1737	0.1515	0.9290
## 3	IPW	2	100	0.0604	0.2382	0.2645	0.2429	1.0154
## 4	Oracle	2	100	0.1510	0.0197	0.0489	0.0035	0.8891
## 5	Naive	2	100	0.0996	0.1459	0.1804	0.1266	0.8637
## 6	IPW	2	100	0.0126	0.2140	0.2693	0.1989	0.9942
## 7	Oracle	3	100	0.1365	0.0144	0.0419	0.0032	0.6612
## 8	Naive	3	100	0.0144	0.2184	0.2336	0.2247	0.8587
## 9	IPW	3	100	0.1976	0.3321	0.3646	0.3554	1.0751
## 10	Oracle	3	100	0.1929	0.0223	0.0537	0.0170	0.7046
## 11	Naive	3	100	0.0182	0.1852	0.2471	0.1998	0.7854
## 12	IPW	3	100	0.1040	0.2924	0.3559	0.3283	1.0021
## 13	Oracle	2	500	0.0172	0.0125	0.0208	0.0038	0.1297
## 14	Naive	2	500	0.1389	0.1475	0.1523	0.1481	0.2064
## 15	IPW	2	500	0.2185	0.2360	0.2362	0.2303	0.3389
## 16	Oracle	2	500	0.0187	0.0136	0.0249	0.0054	0.1379
## 17	Naive	2	500	0.1414	0.1348	0.1483	0.1417	0.2032
## 18	IPW	2	500	0.2111	0.2181	0.2303	0.2267	0.3325

```

## 19 Oracle 3 500 0.0206 0.0151 0.0146 0.0047 0.1152
## 20 Naive 3 500 0.2042 0.2166 0.2231 0.2222 0.3014
## 21 IPW 3 500 0.3194 0.3344 0.3449 0.3445 0.5818
## 22 Oracle 3 500 0.0056 0.0120 0.0217 0.0054 0.1096
## 23 Naive 3 500 0.1936 0.1964 0.2112 0.2080 0.2722
## 24 IPW 3 500 0.3127 0.3172 0.3325 0.3273 0.5367
## 25 Oracle 2 1000 0.0148 0.0079 0.0017 0.0069 0.0666
## 26 Naive 2 1000 0.1704 0.1379 0.1379 0.1435 0.1487
## 27 IPW 2 1000 0.2458 0.2205 0.2200 0.2289 0.2697
## 28 Oracle 2 1000 0.0222 0.0054 0.0045 0.0048 0.0784
## 29 Naive 2 1000 0.1798 0.1314 0.1360 0.1384 0.1492
## 30 IPW 2 1000 0.2645 0.2136 0.2178 0.2203 0.2749
## 31 Oracle 3 1000 0.0262 0.0075 0.0004 0.0059 0.0551
## 32 Naive 3 1000 0.2459 0.2021 0.2057 0.2177 0.2408
## 33 IPW 3 1000 0.3648 0.3205 0.3310 0.3386 0.5137
## 34 Oracle 3 1000 0.0294 0.0070 0.0061 0.0076 0.0615
## 35 Naive 3 1000 0.2437 0.1916 0.1998 0.2029 0.2322
## 36 IPW 3 1000 0.3651 0.3125 0.3220 0.3193 0.4879
## allData[, 15]
## 1 0.3191
## 2 0.3199
## 3 0.3406
## 4 0.3616
## 5 0.3261
## 6 0.3446
## 7 0.2661
## 8 0.2886
## 9 0.3233
## 10 0.2955
## 11 0.2950
## 12 0.3468
## 13 0.0639
## 14 0.0682
## 15 0.0844
## 16 0.0687
## 17 0.0723
## 18 0.0895
## 19 0.0543
## 20 0.0744
## 21 0.1123
## 22 0.0571
## 23 0.0728
## 24 0.1097
## 25 0.0308
## 26 0.0419
## 27 0.0615
## 28 0.0350
## 29 0.0417
## 30 0.0609
## 31 0.0266
## 32 0.0549
## 33 0.0995
## 34 0.0293
## 35 0.0519

```

```
## 36          0.0932
```

```
library(xtable)
```

```
## Warning: package 'xtable' was built under R version 4.1.1
```

```
mseResults <- xtable(mseTable, digits=c(0,0,0,0,4,4,4,4,4,4))
```

```
print(mseResults, include.rownames=FALSE, hline.after=seq(3,24,3))
```

```
% latex table generated in R 4.1.0 by xtable 1.8-4 package % Wed Jan 04 08:54:58 2023
```

Method	m	n	B1	B2	B3	B4	apply(allData[, 11:14], 1, sum)	allData[, 15]
Oracle	2	100	0.1689	0.0141	0.0226	0.0073	0.8341	0.3191
Naive	2	100	0.0452	0.1495	0.1737	0.1515	0.9290	0.3199
IPW	2	100	0.0604	0.2382	0.2645	0.2429	1.0154	0.3406
Oracle	2	100	0.1510	0.0197	0.0489	0.0035	0.8891	0.3616
Naive	2	100	0.0996	0.1459	0.1804	0.1266	0.8637	0.3261
IPW	2	100	0.0126	0.2140	0.2693	0.1989	0.9942	0.3446
Oracle	3	100	0.1365	0.0144	0.0419	0.0032	0.6612	0.2661
Naive	3	100	0.0144	0.2184	0.2336	0.2247	0.8587	0.2886
IPW	3	100	0.1976	0.3321	0.3646	0.3554	1.0751	0.3233
Oracle	3	100	0.1929	0.0223	0.0537	0.0170	0.7046	0.2955
Naive	3	100	0.0182	0.1852	0.2471	0.1998	0.7854	0.2950
IPW	3	100	0.1040	0.2924	0.3559	0.3283	1.0021	0.3468
Oracle	2	500	0.0172	0.0125	0.0208	0.0038	0.1297	0.0639
Naive	2	500	0.1389	0.1475	0.1523	0.1481	0.2064	0.0682
IPW	2	500	0.2185	0.2360	0.2362	0.2303	0.3389	0.0844
Oracle	2	500	0.0187	0.0136	0.0249	0.0054	0.1379	0.0687
Naive	2	500	0.1414	0.1348	0.1483	0.1417	0.2032	0.0723
IPW	2	500	0.2111	0.2181	0.2303	0.2267	0.3325	0.0895
Oracle	3	500	0.0206	0.0151	0.0146	0.0047	0.1152	0.0543
Naive	3	500	0.2042	0.2166	0.2231	0.2222	0.3014	0.0744
IPW	3	500	0.3194	0.3344	0.3449	0.3445	0.5818	0.1123
Oracle	3	500	0.0056	0.0120	0.0217	0.0054	0.1096	0.0571
Naive	3	500	0.1936	0.1964	0.2112	0.2080	0.2722	0.0728
IPW	3	500	0.3127	0.3172	0.3325	0.3273	0.5367	0.1097
Oracle	2	1000	0.0148	0.0079	0.0017	0.0069	0.0666	0.0308
Naive	2	1000	0.1704	0.1379	0.1379	0.1435	0.1487	0.0419
IPW	2	1000	0.2458	0.2205	0.2200	0.2289	0.2697	0.0615
Oracle	2	1000	0.0222	0.0054	0.0045	0.0048	0.0784	0.0350
Naive	2	1000	0.1798	0.1314	0.1360	0.1384	0.1492	0.0417
IPW	2	1000	0.2645	0.2136	0.2178	0.2203	0.2749	0.0609
Oracle	3	1000	0.0262	0.0075	0.0004	0.0059	0.0551	0.0266
Naive	3	1000	0.2459	0.2021	0.2057	0.2177	0.2408	0.0549
IPW	3	1000	0.3648	0.3205	0.3310	0.3386	0.5137	0.0995
Oracle	3	1000	0.0294	0.0070	0.0061	0.0076	0.0615	0.0293
Naive	3	1000	0.2437	0.1916	0.1998	0.2029	0.2322	0.0519
IPW	3	1000	0.3651	0.3125	0.3220	0.3193	0.4879	0.0932

```
coverageTable <- allData[,c(1,3,5,16:19,20:23)]
for(i in 4:ncol(coverageTable)){
  coverageTable[,i] <- round(coverageTable[,i],2)
}
coverageResults <- coverageTable[,1:3]
```

```
for(i in 1:4){
  coverageResults <- cbind(coverageResults, paste0(coverageTable[,i+7], "(", coverageTable[,i+3], ")"))
}

print(xtable(coverageResults, digits=c(0,0,0,0,2,2,2,2)), include.rownames=FALSE, hline.after=seq(3,24,3))
```

% latex table generated in R 4.1.0 by xtable 1.8-4 package % Wed Jan 04 08:54:58 2023

Method	m	n	paste0(coverageTable[, i + 7], "(", coverageTable[, i + 3], ")")	paste0(coverageTable[, i + 7], "(", coverageTable[, i + 3], ")")
Oracle	2	100	0.74(2)	0.79(0.61)
Naive	2	100	0.92(2.5)	0.82(0.79)
IPW	2	100	0.91(2.53)	0.77(0.8)
Oracle	2	100	0.77(2.06)	0.79(0.65)
Naive	2	100	0.9(2.58)	0.86(0.82)
IPW	2	100	0.94(2.61)	0.83(0.84)
Oracle	3	100	0.77(1.63)	0.74(0.51)
Naive	3	100	0.92(2.19)	0.75(0.71)
IPW	3	100	0.92(2.25)	0.55(0.74)
Oracle	3	100	0.74(1.71)	0.74(0.53)
Naive	3	100	0.93(2.28)	0.77(0.73)
IPW	3	100	0.95(2.36)	0.64(0.75)
Oracle	2	500	0.85(0.89)	0.87(0.26)
Naive	2	500	0.88(1.02)	0.53(0.32)
IPW	2	500	0.83(1.04)	0.25(0.33)
Oracle	2	500	0.86(0.94)	0.86(0.28)
Naive	2	500	0.91(1.11)	0.61(0.34)
IPW	2	500	0.88(1.12)	0.34(0.35)
Oracle	3	500	0.76(0.73)	0.72(0.22)
Naive	3	500	0.73(0.89)	0.17(0.28)
IPW	3	500	0.68(0.92)	0.05(0.29)
Oracle	3	500	0.85(0.77)	0.8(0.23)
Naive	3	500	0.85(0.96)	0.29(0.3)
IPW	3	500	0.68(0.98)	0.04(0.31)
Oracle	2	1000	0.79(0.64)	0.88(0.18)
Naive	2	1000	0.77(0.72)	0.35(0.22)
IPW	2	1000	0.67(0.74)	0.01(0.23)
Oracle	2	1000	0.79(0.67)	0.88(0.2)
Naive	2	1000	0.84(0.77)	0.35(0.23)
IPW	2	1000	0.74(0.78)	0.07(0.24)
Oracle	3	1000	0.74(0.52)	0.83(0.15)
Naive	3	1000	0.63(0.63)	0.01(0.2)
IPW	3	1000	0.4(0.65)	0(0.2)
Oracle	3	1000	0.7(0.55)	0.81(0.16)
Naive	3	1000	0.64(0.67)	0.06(0.21)
IPW	3	1000	0.44(0.68)	0(0.21)

Bias and Length